

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
"КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМ. ІГОРЯ СІКОРСЬКОГО"
Факультет електроніки
Кафедра електронної інженерії

"На правах рукопису"

«До захисту допущено»
Завідувач кафедри

УДК _____

_____, В.І. Тимофєєв
" " _____ 20__ р.

Магістерська дисертація

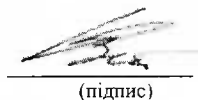
зі спеціальності 153 мікро- та наносистемна техніка

на тему Метод лінеаризації схемних моделей біполярних транзисторів

Виконав: студент 2 курсу, групи ДІМ-21мн
Дроздов Сергій Леонідович
(прізвище, ім'я, по батькові)

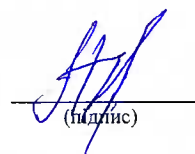

(підпис)

Науковий керівник Доц., к.т.н., Витязь О. О.
(посада, вчене звання, науковий ступінь, прізвище та ініціали)


(підпис)

Консультант _____
(назва розділу) (вчені ступінь та звання, прізвище, ініціали) (підпис)

Рецензент Доц., к.т.н., доц. каф. СП ІІСА Артюхов В.Г.
(посада, вчене звання, науковий ступінь, прізвище та ініціали)


(підпис)

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____
(підпис)

Київ - 2024 року

Національний технічний університет України
“Київський політехнічний інститут ім. Ігоря Сікорського”

Факультет електроніки

Кафедра електронної інженерії

Рівень вищої освіти – другий (магістерський) за освітньо-науковою програмою

Спеціальність 153 мікро- та наносистемна техніка

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ В.І. Тимофєєв

“ ____ ” _____ 20__ р.

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ ДИСЕРТАЦІЮ СТУДЕНТУ

Дроздову Сергію Леонідовичу

(прізвище, ім'я, по батькові)

1. Тема дисертації Метод лінеаризації схемних моделей біполярних транзисторів

Науковий керівник Доц., к.т.н., Витязь О. О.

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від “01” квітня 2024 року № 1518-с

2. Строк подання студентом дисертації “10” червня 2024 року

3. Об'єкт дослідження Біполярні транзистори та їх математичні моделі

4. Предмет дослідження

Метод лінеаризації та адаптації математичних моделей біполярних транзисторів

5. Перелік питань, які потрібно розробити Алгоритм побудови тривимірної поверхні вольт-амперної характеристики (ВАХ) біполярних транзисторів, реалізація алгоритму триангуляції поверхні ВАХ, створення структури бази даних для зберігання точок триангульованої поверхні ВАХ транзисторів, розробка алгоритмів для ефективного пошуку точок у базі даних

6. Перелік графічного (ілюстративного) матеріалу Вольт-амперні характеристики біполярних транзисторів, еквівалентні схеми математичних моделей транзисторів, зображення триангульованих поверхонь сімейства вихідних ВАХ біполярного транзистора, вигляд інтерфейсу користувача

7. Орієнтовний перелік публікацій

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

9. Дата видачі завдання “01” лютого 2024 г.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів роботи	Примітка
1	Огляд літератури і теоретичних основ	01.02.2024 р. – 01.03.2024 р.	Виконано
2	Реалізація алгоритму триангуляції поверхні	01.03.2024 р. – 17.03.2024 р.	Виконано
3	Реалізація алгоритму збереження результатів в базу даних	17.03.2024 р. – 29.03.2024 р.	Виконано
4	Реалізація алгоритму оптимізованого пошуку по утвореним базам даних	29.03.2024 р. – 19.04.2024 р.	Виконано
5	Реалізація методу лінеаризації математичних моделей біполярних транзисторів для підвищення ефективності їх застосування	19.04.2024 р. – 17.05.2024 р.	Виконано
6	Оформлення магістерської дисертації	17.05.2024 р. – 07.06.2024 р.	Виконано

Студент



(підпис)

Дроздов С. Л.

(прізвище та ініціали)

Науковий керівник роботи



(підпис)

Витязь О. О.

(прізвище та ініціали)

РЕФЕРАТ

ЛІНЕАРИЗАЦІЯ, МАТЕМАТИЧНІ МОДЕЛІ, БІПОЛЯРНІ ТРАНЗИСТОРИ, ВОЛЬТ-АМПЕРНІ ХАРАКТЕРИСТИКИ, ТРИАНГУЛЯЦІЯ ПОВЕРХНІ

Метою даної магістерської дисертації є розробка методу лінеаризації математичних моделей біполярних транзисторів для підвищення ефективності їх застосування в системах автоматизованого проектування електронних кіл. Для контролю точності апроксимації розроблена процедура адаптивного керування триангуляцією поверхні, яка є геометричним образом математичної моделі транзистора. В роботі досліджується застосування математичних моделей для побудови вольт-амперних характеристик (ВАХ) біполярних транзисторів, зокрема, на основі моделі Еберса-Молла. Об'єктом дослідження є біполярні транзистори та їх математичні моделі. Предметом дослідження є метод лінеаризації та адаптації математичних моделей біполярних транзисторів для їх застосування в системах автоматизованого схемотехнічного аналізу в електроніці.

Основні завдання дослідження включають розробку алгоритму обчислення, побудови та триангуляції тривимірної поверхні ВАХ, створення структури бази даних для зберігання точок триангульованої поверхні ВАХ транзисторів, розробку алгоритмів для ефективного пошуку точок у базі даних, а також проведення тестування системи на різних моделях біполярних транзисторів.

В результаті виконання зазначених завдань розроблено тестову програмну систему для розрахунку ВАХ біполярних транзисторів, яка забезпечує автоматизовану адаптацію лінеаризованих моделей транзистора до критеріїв точності. Система є засобом підвищення ефективності систем автоматизованого проектування у галузі електроніки за рахунок скорочення тривалості етапу схемотехнічного аналізу. Проведені експерименти підтверджують ефективність запропонованого методу лінеаризації та перспективність його практичного застосування для проектування електронних пристроїв.

ABSTRACT

LINEARIZATION, MATHEMATICAL MODELS, BIPOLAR TRANSISTORS, CURRENT-VOLTAGE CHARACTERISTICS, SURFACE TRIANGULATION

The aim of this master's thesis is to develop a method for linearizing mathematical models of bipolar transistors to enhance their efficiency in electronic circuit design automation systems. To control the approximation accuracy, an adaptive surface triangulation management procedure has been developed, which serves as the geometric representation of the transistor's mathematical model. The study investigates the application of mathematical models to construct the current-voltage characteristics (IVC) of bipolar transistors, particularly based on the Ebers-Moll model. The research object is bipolar transistors and their mathematical models. The subject of the research is the method of linearizing and adapting mathematical models of bipolar transistors for use in automated electronic circuit analysis systems.

The main objectives of the research include the development of an algorithm for calculating, constructing, and triangulating a three-dimensional IVC surface, creating a database structure for storing triangulated surface points of the transistors' IVC, developing algorithms for efficient point retrieval in the database, and conducting system testing on various models of bipolar transistors.

As a result of these tasks, a test software system for calculating the IVC of bipolar transistors was developed, which provides automated adaptation of the linearized transistor models to accuracy criteria. The system enhances the efficiency of electronic circuit design automation systems by reducing the duration of the circuit analysis stage. The experiments conducted confirm the effectiveness of the proposed linearization method and its potential practical application in electronic device design.

ЗМІСТ

СКРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ	8
ВСТУП	9
1 ОГЛЯД ЛІТЕРАТУРИ І ТЕОРЕТИЧНИХ ОСНОВ	11
1.1 Основи роботи біполярного транзистора	11
1.2 Вольт-амперні характеристики (ВАХ) біполярного транзистора.....	18
1.2.1 Характеристики для схеми із загальною базою.....	21
1.2.2 Характеристики для схеми із загальним емітером.....	25
1.3 Математичні моделі біполярного транзистора	29
1.3.1 Математична модель Еберса-Молла.....	30
1.3.2 Математична модель Гуммеля-Пуна	33
1.3.3 Математична модель переносу.....	35
1.4 Висновки	37
2 ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИКА ДОСЛІДЖЕННЯ.....	39
2.1 Формулювання задачі	39
2.2 Методи триангуляції в комп'ютерній графіці та чисельних розрахунках.....	40
2.2.1 Триангуляція Делоне	41
2.2.2 Алгоритм Рупперта.....	42
2.2.3 Алгоритм Боулера-Вотсона	43
2.2.4 Grid-Based триангуляція.....	43
2.2.5 Edge Subdivision	44
2.2.6 Midpoint Subdivision.....	44
2.2.7 Centroid Subdivision	45
2.2.8 Застосування триангуляції.....	46
2.3 Програмні засоби і середовища розробки	46
2.3.1 Мови програмування	47
2.3.2 Бази даних.....	50
2.3.3 Бібліотеки та інструменти.....	53
2.4 Опис бази даних та її структури	54
2.5 Методи оптимізованого пошуку в базах даних	57
2.6 Висновки	60
3 РОЗРОБКА СИСТЕМИ	61
3.1. Архітектура програмної системи	61
3.2 Вибір алгоритму триангуляції поверхні ВАХ.....	62

3.3 Розробка алгоритму для введення даних для побудови ВАХ	70
3.4 Збереження точок поверхні в базу даних.	71
3.5 Оптимізація пошуку за двома координатами для визначення третьої із наперед заданою точністю	72
3.6 Висновки	74
4 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ	76
4.1 Програмна реалізація системи.....	76
4.2. Приклади введення формул ВАХ та отримані результати.....	81
4.3 Оцінка точності і швидкодії пошуку в базі даних.....	83
4.4 Висновки	85
5 АНАЛІЗ ТА ОБГОВОРЕННЯ РЕЗУЛЬТАТІВ	86
5.1 Аналіз ефективності алгоритмів.....	86
5.2 Вплив точності триангуляції на результати та тривалість обчислень.....	87
5.3 Обговорення вдосконалення системи та її застосування.....	90
5.4 Висновки	91
ВИСНОВКИ.....	93
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	94
ДОДАТОК А.1	97
ДОДАТОК А.2.....	97
ДОДАТОК А.3	99

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

БТ – біполярний транзистор

ВАХ – вольт-амперна характеристика

Б - база

Е - емітер

К – колектор

СУБД - системи управління базами даних

РБД - реляційні бази даних

SQL (Structured Query Language) - структурована мова запитів

UI (User Interface) - інтерфейс користувача

ВСТУП

Удосконалення моделей нелінійних електронних компонентів, до складу яких входять і біполярні транзистори, є актуальною проблемою, оскільки вимоги до точності моделі і ефективності їх застосування є такими, які протирічать одна одній, що вимагає пошуку прийняттого компромісу. Незважаючи на значний прогрес у розробці моделей напівпровідникових приладів, ефективність їх застосування залишається одним із ключових завдань. Наприклад, такі моделі як Еберса-Молла та Гуммеля-Пуна, забезпечують прийнятну точність, але їх складність зменшує ефективність їх застосування у системах автоматизованого проектування електронних кіл, оскільки потребує великої кількості обчислень. Ця обставина вимагає розробки нових підходів, наприклад, заснованих на лінеаризації математичних моделей.

Сучасні пакети автоматизованого схемотехнічного проектування мають у своєму складі бібліотеки моделей нелінійних електронних компонентів, застосування яких у математичних моделях кіл, що проектуються, потребує великого об'єму обчислювальних витрат саме на обчислення нелінійних функцій, які входять до складу моделей компонентів, у тому числі і біполярних транзисторів. Крім того, розвиток нових технологій і підвищення вимог до характеристик електронних пристроїв вимагають розробки більш точних, а, значить, і ще більш складних моделей компонентів.

У даній роботі розглядаються проблеми лінеаризації математичних моделей біполярних транзисторів з використанням триангуляції поверхонь, які є графічними образами їх вольт-амперних характеристик. Запропонований спосіб дозволяє підвищити ефективність та швидкодію аналізу, що є важливим для проектування сучасних електронних пристроїв. Реалізація алгоритму побудови тривимірної поверхні ВАХ, триангуляція поверхні ВАХ та створення структури бази даних для

зберігання точок триангульованої поверхні забезпечують комплексний підхід до вирішення проблеми лінеаризації.

Запропонована система буде корисною для інженерів та науковців, які займаються розробкою і вдосконаленням систем автоматизованого проектування, і спрямована на забезпечення більш ефективного застосування математичних моделей біполярних транзисторів для розв'язування задач схемотехнічного аналізу.

1 ОГЛЯД ЛІТЕРАТУРИ І ТЕОРЕТИЧНИХ ОСНОВ

1.1 Основи роботи біполярного транзистора

Біполярний транзистор (БТ) –це напівпровідниковий приладом, який є одним з основних складових сучасної електроніки і використовується як дискретний елемент електронних пристроїв чи як елемент інтегральних мікросхем. У мікросхемах об'єднана велика кількість інтегрованих біполярних транзисторів в загальний функціональний блок [2]. Принцип роботи БТ базується на взаємодії двох електронно-діркових переходів, де носіями заряду є електрони, що мають негативний заряд та вільно рухаються в кристалічній решітці та відповідають за провідність матеріалу, а також переходять з валентної зони в зону провідності під впливом зовнішнього електричного поля, та дірки (фактично "вільне місце" для електрона) — позитивно заряджені квазічастинки, які виникають внаслідок відсутності електрона в кристалічній решітці та можуть рухатися від одного атома до іншого, викликаючи ефективний рух позитивно заряджених частинок. Призначення біполярного транзистора полягає в підсиленні та генеруванні електричних сигналів.

Біполярний транзистор складається з трьох областей р- і n-типу, що чергуються і формують електронно-діркові переходи. На рисунку 1.1 представлено структуру і умовне позначення транзистора. Виділяють транзистори типів р-n-p і n-p-n в залежності від провідності середньої та крайніх зон. Різниця принципу роботи р-n-p і n-p-n типів полягає в полярності живлення транзистора [1]. Середня зона транзистора називається базою (Б). Одна з крайніх зон, призначена для введення носіїв у базу, називається емітером (Е), а інша, що збирає носії з бази, називається колектором (К). Переходи між емітером і базою та між колектором і базою називаються відповідно емітерним і колекторним переходами.

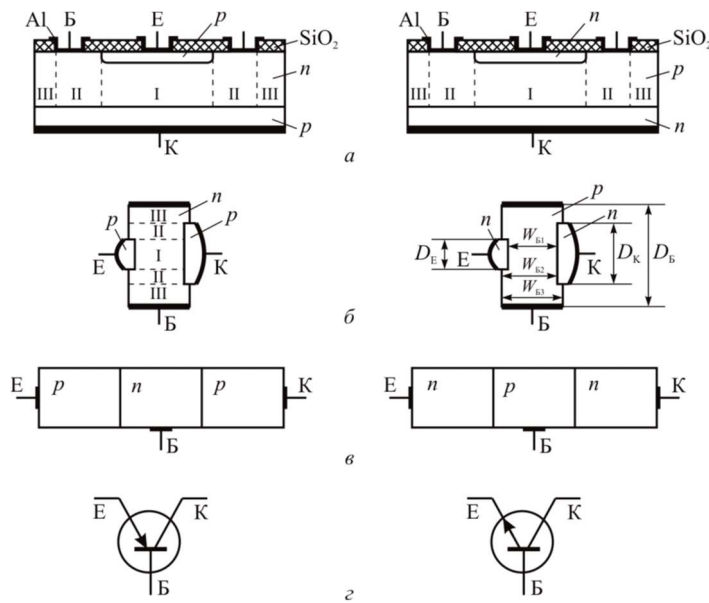


Рисунок 1.1 - Структура дифузійного транзистора (а) та сплавного транзистора (б), плоска модель транзисторів р-п-р і п-р-п типу (в) та їх умовні позначення (г) [1]

Емітер є шаром, що інжектує носії заряду (електрони або дірки) в базу. Цей шар є сильно легований для забезпечення високої концентрації носіїв. База - це тонкий середній шар з меншою концентрацією домішок, що контролює потік носіїв заряду між емітером і колектором. Колектор - це шар, який збирає носії заряду, що проходять через базу [3]. Він має більшу площу та менш сильну легованість порівняно з емітером.

Але є різниця і окремих частин шарів, наприклад роль окремих частин бази в роботі реального транзистора не однакова. Частину бази, через яку рухаються носії до колектора і що розміщена безпосередньо під емітером, називають активною базою; частину бази, що розміщена між емітером і базовим виводом, називають пасивною, а частину, що лежить за базовим виводом, — периферійною (рис. 1.1, а). На рисунку 1.1 (б) зображено сплавлений транзистор, частина I якого є активною базою даного транзистора, частина II - пасивною, частина III - периферійною. Для ефективної взаємодії р-п переходів товщина першої частини бази (активної) транзистора має бути набагато меншою від дифузійної довжини неосновних носіїв [1].

Сучасні біполярні транзистори найчастіше виготовляються на основі кремнію за дифузійною технологією. Ця технологія дозволяє сформувати в початковій пластині р-типу два р-п переходи методом послідовної дифузії донорних і

акцепторних домішок. Таким чином утворюється транзистор р-п-р типу, а якщо початкова пластина є п-типу, то утворюється транзистор п-р-п типу (рис 1.1, в).

Глибина залягання р-п переходів та товщина бази (що дорівнює декільком мікрометрам) визначається тривалістю дифузії [1]. На рисунку 1.2 (а) можна побачити профіль розподілу домішок N_A та N_D і результуючий розподіл $N_A - N_D$.

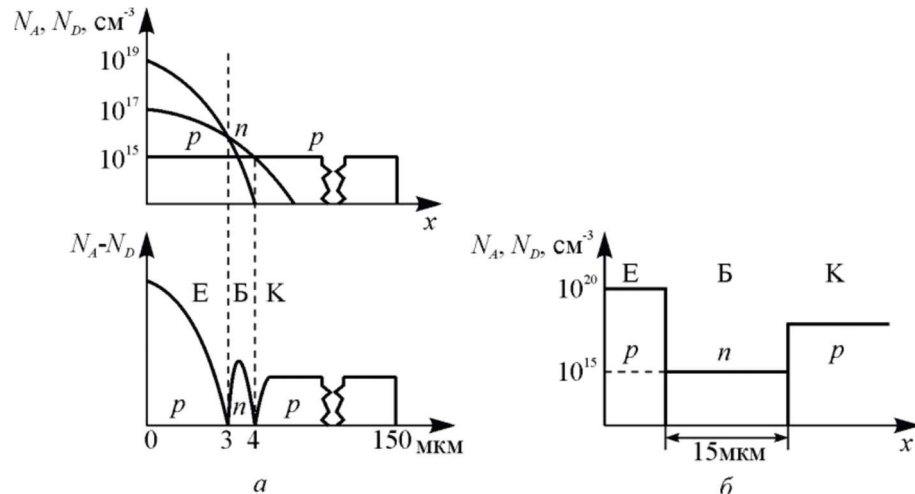


Рисунок 1.2 - Профілі розподілу домішок у двох типах транзисторів: дифузійних (а) та сплавних (б) [1]

Властивості транзисторів і їхні характеристики значною мірою залежать від перенесення носіїв заряду між емітером та колектором, що визначається розподілом домішок у базі. У рівноважному стані концентрація носіїв у транзисторі визначається законом діючих мас: $np = n_i^2$. Концентрація основних носіїв залежить від ступеня легування кожної області. Найчастіше, в області бази бездрейфового транзистора концентрація домішок на два-три порядки менша, ніж у емітері та колекторі.

При прикладанні напруги зміщення на р-п перехід концентрація неосновних носіїв змінюється: за прямого зміщення вона зростає, а за зворотного — зменшується. Відповідно змінюється й концентрація основних носіїв біля межі р-п переходів у транзисторі.

Існує чотири режими роботи транзистора залежно від знаків напруг на емітерному та колекторному переходах [5]:

- Режим відсікання: обидва переходи зміщені у зворотному напрямі, струм мінімальний.

- Режим насичення: обидва переходи зміщені у прямому напрямі, транзистор працює як закрите з'єднання, напруга між колектором і емітером зменшується.

- Активний режим (режим підсилення): емітерний перехід зміщений у прямому напрямі, колекторний — у зворотному. Емітер інjektує носії заряду в базу, де більшість переноситься до колектора, створюючи підсилення струму.

- Зворотний активний режим: емітерний перехід зміщений у зворотному напрямі, колекторний — у прямому. Використовується рідко через гірші характеристики підсилення.

Концентрації неосновних носіїв на межі областей транзистора залежать від прикладеної напруги і розподілу домішок у базі. У загальному випадку концентрації неосновних носіїв на межі областей транзистора можна подати [1]:

1. Область емітера:

- при $x = -l_E$, $n_{pE}(-l_E) = n_{p_0E}$;
- при $x = 0$, $n_{pE}(0) = n_{p_0E} e^{\frac{U_{EB}}{\varphi_T}}$.

2. Область бази:

- при $x = 0$, $p_{nB}(0) = p_{n_0B} e^{\frac{U_{EB}}{\varphi_T}}$;
- при $x = l_{BK}$, $p_{nB}(l_{BK}) = p_{n_0B} e^{\frac{U_{BK}}{\varphi_T}}$.

3. Область колектора:

- при $x = l_{BK}$, $n_{pK}(l_{BK}) = n_{p_0K} e^{\frac{U_{BK}}{\varphi_T}}$;
- при $x = l_K$, $n_{pK}(l_K) = n_{p_0K}$.

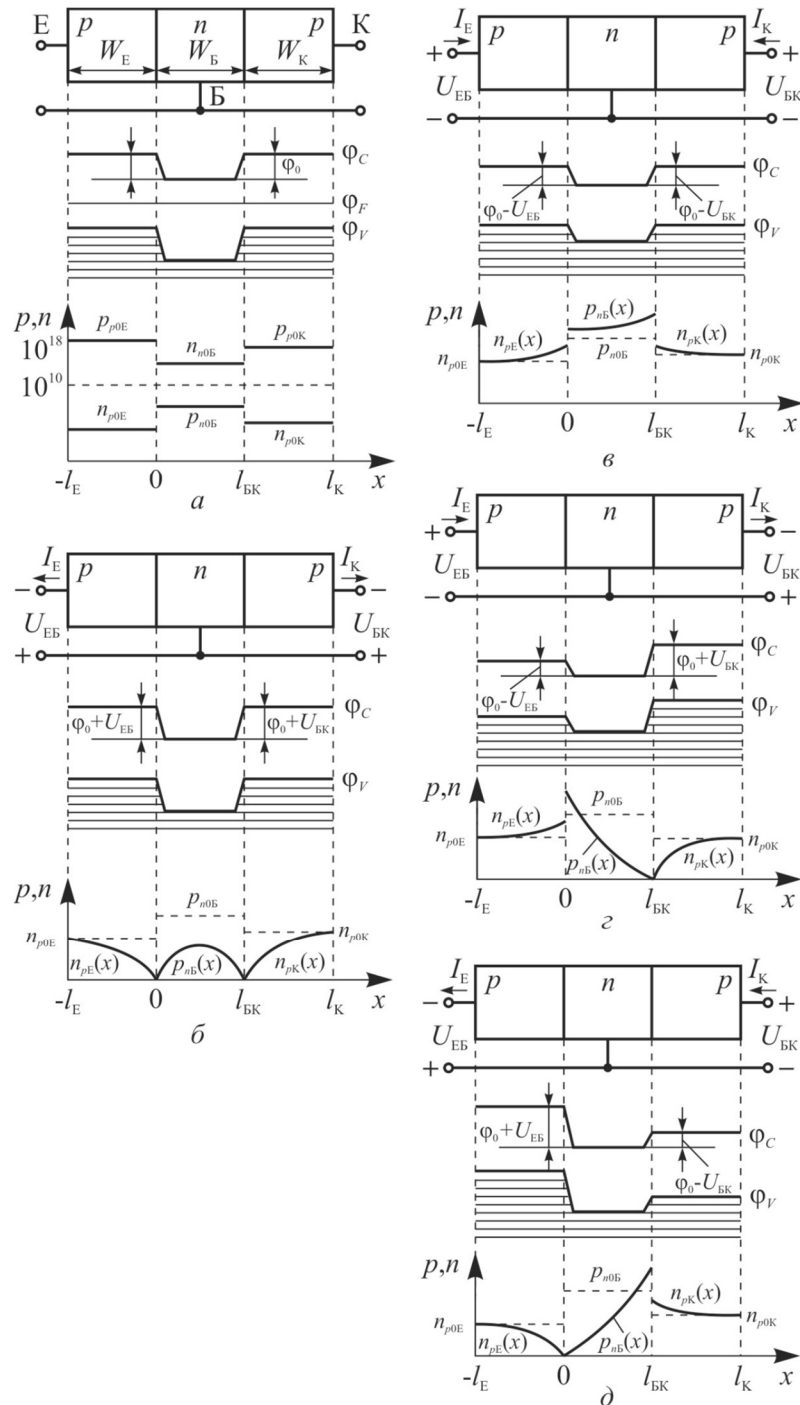


Рисунок 1.3 - Зображення енергетичних діаграм та розподілу неосновних носіїв заряду (електронів і дірок) в різних областях транзистора в п'яти різних режимах роботи: в рівноважному стані (а), у режимі відсікання (б), у режимі насичення (в), в активному нормальному режимі (г) та в активному інверсному режимі (д) [1]

Відстань від $x = 0$ до $x = -l_K$ - товщина емітера W_E ;

відстань від $x = 0$ до $x = l_{BK}$ - товщина бази W_B ;

відстань від $x = l_{BK}$ до $x = l_K$ - товщина колектора W_K ;

U_{EB} - напруга між емітером і базою;

U_{BK} - напруга між колектором і базою.

Розподіл надмірних концентрацій носіїв у транзисторі мають наступний вигляд:

- Для емітера

$$\Delta n_{pE} = n_{p0E} \left(e^{\frac{U_{EB}}{\varphi_T}} - 1 \right) \frac{\text{sh} \frac{W_E - x}{L_n}}{\text{sh} \frac{W_E}{L_n}} \Big|_{-l_E \leq x \leq 0} \quad (1.1)$$

- Для колектора

$$\Delta n_{pK} = n_{p0K} \left(e^{\frac{U_{BK}}{\varphi_T}} - 1 \right) \frac{\text{sh} \frac{l_K - x}{L_n}}{\text{sh} \frac{W_K}{L_n}} \Big|_{-l_{BK} \leq x \leq l_K} \quad (1.2)$$

- Для бази

$$\Delta p_{nB} = p_{n0B} \left(e^{\frac{U_{EB}}{\varphi_T}} - 1 \right) \frac{\text{sh} \frac{l_{BK} - x}{L_p}}{\text{sh} \frac{W_B}{L_p}} + p_{n0B} \left(e^{\frac{U_{BK}}{\varphi_T}} - 1 \right) \frac{\text{sh} \frac{x}{L_p}}{\text{sh} \frac{W_B}{L_p}} \Big|_{0 \leq x \leq l_{BK}} \quad (1.3)$$

У формулах (1.1) - (1.3) L_n, L_p - довжина дифузії електронів і дірок відповідно

Основні фізичні процеси, що визначають роботу біполярного транзистора, включають інжекцію носіїв заряду, дифузію, рекомбінацію та перенесення заряду [7].

1. Інжекція носіїв заряду відбувається, коли емітер транзистора n-p-n або транзистора p-n-p викидає електрони (у транзисторі n-p-n) або дірки (у транзисторі p-n-p) в базу, що забезпечує основний струм через транзистор.

2. Дифузія (рух носіїв заряду) - це процес, при якому носії заряду, введені в базу транзистора з емітера за допомогою емітерного струму, дифундують (переміщуються) через неї, генеруючи базовий струм. Тонка основа транзистора мінімізує втрату носіїв заряду через рекомбінацію (зустріч і зникнення носіїв заряду протилежного знаку), забезпечуючи більш ефективне посилення сигналу.

3. Рекомбінація полягає в тому, що деякі з носіїв заряду рекомбінують (два носії заряду протилежного знаку (електрон і дірка) зустрічаються і зникають, вивільняючи енергію у вигляді тепла або фотона) у базі, що визначає базовий струм. Зазвичай цей струм дуже малий порівняно зі струмами випромінювання та збору.

4. Дрейф - це явище переміщення електричних носіїв під дією електричного поля до колектора, що призводить до створення колекторного струму.

Статичні та динамічні характеристики біполярного транзистора використовуються для аналізу його працездатності. До статичних характеристик відносяться вольт-амперні характеристики (ВАХ), які відображають залежність струмів емітера, бази і колектора від прикладених напруг.

1. Характеристики передачі струму - залежність струму колектора від струму бази.

2. Вхідні характеристики - залежність струму бази від напруги база-емітер

3. Вихідні характеристики - залежність струму колектора від напруги колектор-емітер при різних значеннях струму бази.

Динамічні характеристики враховують частотні характеристики транзистора, особливо здатність передавати високочастотні сигнали. Вони визначаються такими параметрами, як частота зрізу, яка описує швидкість транзистора.

1.2 Вольт-амперні характеристики (ВАХ) біполярного транзистора

Статичні характеристики відображають основний зв'язок між струмами та напругами в транзисторі. Аналітичні рівняння, що описують взаємозв'язок струмів і напруг у транзисторі, можна отримати на основі розподілу неосновних носіїв у базі (1.3), диференціюючи це рівняння по x можна знайти значення діркового струму бази [1]:

$$I_p(x) = -qD_p \frac{d\Delta p_n}{dx} S = \quad (1.4)$$

$$= qD_p \frac{p_{n0B}}{\operatorname{sh} \frac{W_B}{L_p}} S \left[\left(e^{\frac{U_{EB}}{\varphi_T}} - 1 \right) \operatorname{ch} \frac{W_B - x}{L_p} - \left(e^{\frac{U_{BK}}{\varphi_T}} - 1 \right) \operatorname{ch} \frac{x}{L_p} \right].$$

Припустимо, що в цьому рівнянні $x = 0$ і $x = W_B$. Тоді знайдемо діркові складові струму емітера I_{pE} і струму колектора I_{pK} :

$$I_{pE} = \frac{qD_p}{L_p} \frac{p_{n0B}}{\operatorname{sh} \frac{W_B}{L_p}} S_E \left[\left(e^{\frac{U_{EB}}{\varphi_T}} - 1 \right) \operatorname{ch} \frac{W_B}{L_p} - \left(e^{\frac{U_{BK}}{\varphi_T}} - 1 \right) \right]; \quad (1.5)$$

$$I_{pK} = \frac{qD_p}{L_p} \frac{p_{n0B}}{\operatorname{sh} \frac{W_B}{L_p}} S_K \left[\left(e^{\frac{U_{EB}}{\varphi_T}} - 1 \right) - \left(e^{\frac{U_{BK}}{\varphi_T}} - 1 \right) \operatorname{ch} \frac{W_B}{L_p} \right], \quad (1.6)$$

де S_E, S_K - площі емітерного та колекторного переходів; D_p, L_p - коефіцієнт дифузії та довжина дифузії дірок у базі.

Формули (1.5) та (1.6) для активного режиму транзистора (коли $|U_{BK}| > \varphi_T$) можна подати в наступному вигляді:

$$I_{pE} = q \frac{D_p}{L_p} \frac{p_{n0B}}{\operatorname{sh} \frac{W_B}{L_p}} S \left[\left(e^{\frac{U_{EB}}{\varphi_T}} - 1 \right) \operatorname{ch} \frac{W_B}{L_p} + 1 \right]; \quad (1.7)$$

$$I_{pK} = q \frac{D_p}{L_p} \frac{p_{n0B}}{\operatorname{sh} \frac{W_B}{L_p}} S \left[\left(e^{\frac{U_{EB}}{\varphi_T}} - 1 \right) + \operatorname{ch} \frac{W_B}{L_p} \right]. \quad (1.8)$$

Електронна складова емітерного струму визначається диференціюванням рівняння (1.1) по x , а також, якщо $W_E \gg L_n$ і $x = 0$:

$$I_{nE} = qD_n \frac{d\Delta n_{pE}}{dx} S = q \frac{D_n}{L_n} n_{p0E} S \left(e^{\frac{U_{EB}}{\varphi_T}} - 1 \right), \quad (1.9)$$

де D_n, L_n - коефіцієнт дифузії та довжина дифузії електронів у емітері.

Повний струм через емітерний перехід за формулами (1.7) та (1.9) має наступний вигляд:

$$I_E = \left(q \frac{D_p}{L_p} p_{n0B} \operatorname{cth} \frac{W_E}{L_p} + q \frac{D_n}{L_n} n_{p0E} \right) S \left(e^{\frac{U_{EB}}{\varphi_T}} - 1 \right) + q \frac{D_p}{L_p} p_{n0B} S \frac{1}{\operatorname{sh} \frac{W_E}{L_p}}. \quad (1.10)$$

Подібним чином можна визначити електронну компоненту колекторного струму. Електронні складові струмів емітера та колектора значно менші за діркові складові і часто можуть бути знехтувані у загальних струмах через переходи.

Сім'ї емітерних і колекторних характеристик транзистора задані рівняннями (1.5) і (1.6). Перепишемо ці рівняння у наступному вигляді:

$$I_{pE} = I_{EP} \left(e^{\frac{U_{EB}}{\varphi_T}} - 1 \right) - I_{KB0} \left(e^{\frac{U_{BK}}{\varphi_T}} - 1 \right), \quad (1.11a)$$

$$I_{pK} = I_{EB0} \left(e^{\frac{U_{EB}}{\varphi_T}} - 1 \right) - I_{KP} \left(e^{\frac{U_{BK}}{\varphi_T}} - 1 \right). \quad (1.11b)$$

Розпишемо складові формул (1.11):

$I_{EP} = q \frac{D_p}{L_p} p_{n0B} \operatorname{cth} \frac{W_E}{L_p} S$ - початковий (зворотний) струм колектора при короткому замиканні в колі емітера;

$I_{KB0} = q \frac{D_p}{L_p} \frac{p_{n0B}}{\operatorname{sh} \frac{W_E}{L_p}} S$ - тепловий (зворотний) струм емітера при холостому ході в колі колектора;

$$I_{\text{ЕБ}0} = q \frac{D_p}{L_p} \frac{p_{n_0 \text{Б}}}{\text{sh} \frac{W_{\text{Б}}}{L_p}} S - \text{тепловий (зворотний) струм колектора при холостому ході}$$

в колі емітера;

$$I_{\text{Кп}} = q \frac{D_p}{L_p} p_{n_0} \text{cth} \frac{W_{\text{Б}}}{L_p} S - \text{початковий (зворотний) струм емітера при короткому}$$

замиканні в колі колектора.

Емітерна характеристика при $U_{\text{БК}} = 0$ (рис. 1.4, а) відповідає прямій гілці типової діодної характеристики. Якщо $U_{\text{БК}} > 0$, криві зміщуються вправо і вниз через збільшення емітерного струму (друга складова рівняння (4.11, а)). При $U_{\text{БК}} < 0$ криві зсуваються трохи вліво і вгору. Коли $|U_{\text{БК}}| \gg \varphi_T$, колекторна напруга майже не впливає на характеристику. Набір колекторних характеристик зазвичай зображують у першому квадранті, хоча $U_{\text{БК}}$ має від'ємні значення (рис. 1.4, б).

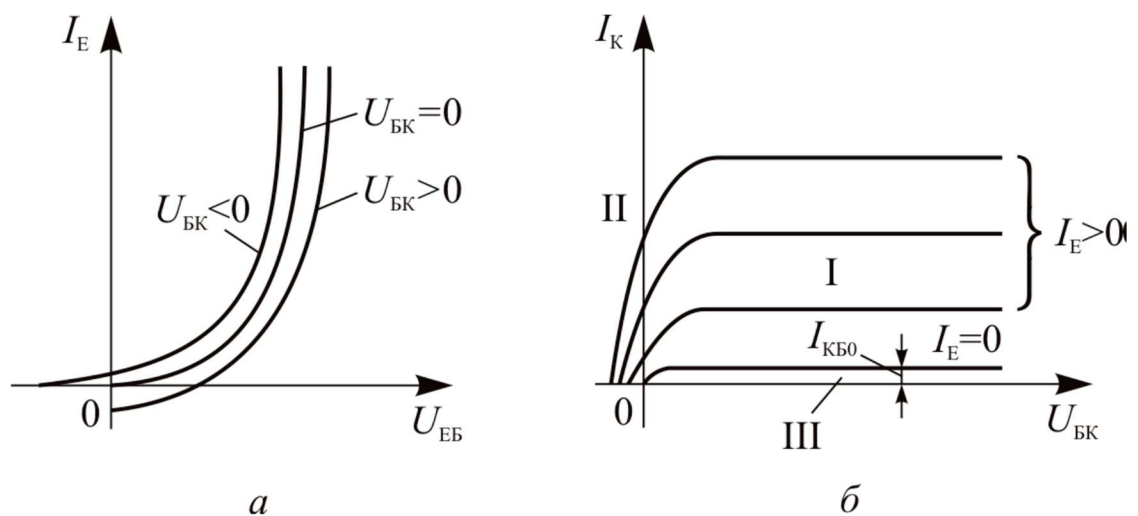


Рисунок 1.4 - Сім'ї статичних вольт-амперних характеристик емітерних (а) та колекторних (б) ідеалізованого транзистора [1]

Статичні характеристики справжнього транзистора відрізняються від характеристик ідеалізованого аналога з тих самих причин, що й параметри реального р-п переходу. Крім цього, на характеристики транзистора впливають ефект Ерлі та зміни параметрів в залежності від режиму роботи. Залежно від того, які струми та напруги обираються як незалежні змінні, а які як залежні, для кожної схеми

включення можливі різні варіанти функціональних зв'язків і відповідні їм набори статичних характеристик.

Зазвичай використовують групи вхідних і вихідних характеристик, в яких вхідна і вихідна напруги виступають як незалежні змінні величини

$$[1]: \quad I_{BX} = f(U_{BX})|_{U_{BIX} = \text{const}}; \quad (1.12)$$

$$I_{BIX} = f(U_{BIX})|_{I_{BX} = \text{const}}. \quad (1.13)$$

Характеристика прямої передачі є похідною (1.12):

$$I_{BIX} = f(I_{BX})|_{U_{BIX} = \text{const}}. \quad (1.14)$$

Характеристика зворотного зв'язку є похідною (1.13):

$$U_{BX} = f(U_{BIX})|_{I_{BIX} = \text{const}}. \quad (1.15)$$

Форма вольт-амперних характеристик (ВАХ) залежить від схеми підключення транзистора. Зазвичай аналізують характеристики для схем із загальною базою та загальним емітером. Варто зазначити, що характеристики для схеми із загальним колектором практично не відрізняються від характеристик схеми із загальним емітером.

1.2.1 Характеристики для схеми із загальною базою

Вхідні характеристики описують як струм емітера змінюється залежно від напруги між емітером і базою при постійній напрузі на колекторі

$$[1]: \quad I_E = f(U_{EB})|_{U_{BK} = \text{const}}. \quad (1.16)$$

Якщо $U_{BK} = 0$, то вхідна характеристика повністю збігається з ВАХ реального р-п переходу, зміщеного в прямому напрямку (рис. 1.5, а). При подачі негативної напруги на колекторний перехід вхідна ВАХ зсувається ліворуч. Це зміщення пояснюється наявністю зворотного зв'язку в транзисторі, де опір бази є спільним для

вхідного та вихідного кіл. Зворотний зв'язок діє наступним чином - збільшення напруги $|U_{БК}|$ призводить або до зростання емітерного струму I_E за постійної напруги емітер - база, або до її зменшення при постійному емітерному струмі.

Відстань між характеристиками при однаковому прирості $|U_{БК}|$ буде різною, оскільки коефіцієнт зворотного зв'язку зменшується зі збільшенням цієї напруги. При великих значеннях $|U_{БК}|$ вхідні характеристики зливаються в одну.

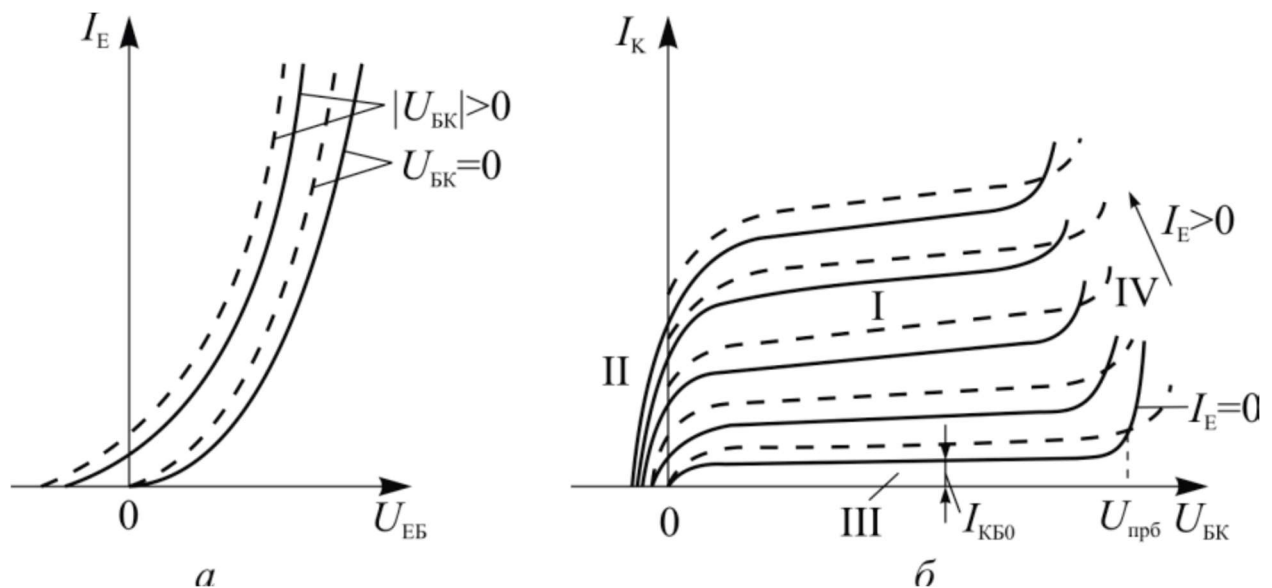


Рисунок 1.5 – Вхідні (а) вольт-амперні характеристики транзистора у схемі із загальною базою та вихідні (б) ($-T = +20\text{ }^{\circ}\text{C}$; $--T = +50\text{ }^{\circ}\text{C}$) [1]

Вихідні характеристики відображають залежність струму колектора від напруги між базою та колектором за умови постійного струму емітера:

$$I_K = f(U_{БК})|_{I_E = \text{const}} \quad (1.17)$$

Також можна наближено описати вихідні характеристики наступною формулою:

$$I_K = \alpha I_E - I_{КБ0} \left(e^{\frac{U_{БК}}{\varphi_T}} - 1 \right) \quad (1.18)$$

Якщо струм емітера I_E дорівнює нулю, а на колекторний перехід подається зворотне зміщення, то вихідна характеристика аналогічна зворотній гілці ВАХ реального р-п переходу (рис. 1.5, б). Коли задається струм I_E , вихідні характеристики зміщуються вгору на αI_E .

Невеликий нахил характеристик відносно осі абсцис обумовлений залежністю коефіцієнта підсилення α і струму I_{KB0} від напруги U_{BK} . Залежність α від струму емітера I_E призводить до того, що при однаковому прирості струму емітера відстані між характеристиками будуть різними.

Початковий відрізок характеристик зсунутий від початку координат у напрямку додатних напруг U_{BK} . Якщо $U_{BK} = 0$ і $I_E > 0$, у колі колектора протікає початковий струм. Для того, щоб звести цей струм до нуля, необхідно змістити колекторний перехід у прямому напрямку.

На сім'ї вихідних (колекторних) характеристик можна виділити чотири області, які відповідають різним режимам роботи транзистора (рис. 1.5, б):

- Область I знаходиться у першому квадранті вище струму I_{KB0} і відповідає активному режиму роботи. У цій області підсилення електричних сигналів відбувається майже без нелінійних спотворень, що робить її основною для підсилювальних схем.
- Область II розташована у другому квадранті і відповідає режиму насичення, коли обидва переходи зміщені у прямому напрямку. За певних умов транзистор може перейти з активного режиму в режим насичення без зміни полярності напруги джерела живлення. Під час протікання колекторного струму $I_K = \alpha I_E$ на резисторі R_H виділяється напруга U_H , полярність якої протилежна до напруги джерела живлення U_K . Це зменшує напругу на колекторному р-п переході. Зі збільшенням I_E напруга U_H стає більшою за напругу U_K , колекторний р-п перехід зміщується у прямому напрямку і транзистор входить у режим насичення. У режимі насичення транзистор повністю відкритий і має мінімальний опір [8].

- Область III розташована у першому квадранті нижче струму $I_{КБ0}$ і відповідає режиму відсікання. У цьому режимі транзистор повністю закритий і його опір максимальний.
- Область IV відповідає пробою колекторного переходу. В цій області, через лавинне помноження носіїв заряду, струм I_K різко зростає з підвищенням напруги $U_{БК}$. Ця область небезпечна для роботи, оскільки лавинний пробій може перейти у вторинний пробій, що призводить до виходу транзистора з ладу.

Активний режим є основним для підсилювальних схем, тоді як режими насичення та відсікання характерні для ключових імпульсних схем.

Характеристики прямої передачі визначають залежність струму колектора від струму емітера при постійній напрузі на колекторі:

$$I_K = f(I_E) | U_{ЕК} = \text{const} \cdot \quad (1.19)$$

Ці характеристики мають вигляд прямих ліній, але внаслідок залежності коефіцієнта α від струму I_E характеристики мають певні відхилення від лінійного закону (рис. 1.6,а). Зміщення характеристик вгору при підвищенні напруги на колекторі відбувається через збільшення струму $I_{КБ0}$ і коефіцієнта α .

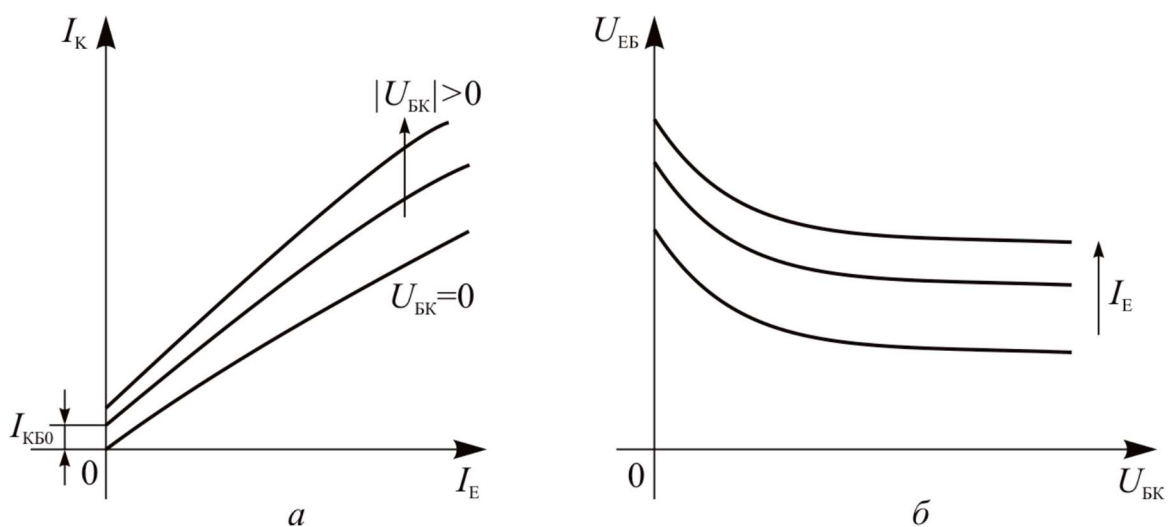


Рисунок 1.6 - Вольт – амперні характеристики прямої передачі (а) та зворотного зв'язку (б) для транзистора у схемі із загальною базою [1]

Характеристики зворотного зв'язку показують залежність напруги на емітерному переході від напруги на колекторі при постійному струмі емітера:

$$U_{EB} = f(U_{BK})|_{I_E = \text{const}}. \quad (1.20)$$

Ця залежність обумовлена наявністю внутрішнього зворотного зв'язку в транзисторі. При збільшенні напруги U_{BK} зменшується напруга U_{EB} , і ця залежність є нелінійною, оскільки з підвищенням U_{BK} зворотній зв'язок у транзисторі послаблюється (рис. 1.6, б). При збільшенні струму I_E характеристики зміщуються вгору, оскільки збільшується прямий спад напруги на емітерному переході.

1.2.2 Характеристики для схеми із загальним емітером

У схемі із загальним емітером входними характеристиками є залежність струму бази від напруги на емітерному переході при постійній напрузі на колекторі [1]:

$$I_B = f(U_{EB})|_{U_{EK} = \text{const}}. \quad (1.21)$$

На рисунку 1.7 (а) можна побачити характер цієї залежності, що схожий на характер входних характеристик для схеми із загальною базою. Струм бази складається з двох складових: рекомбінаційної складової I_{BK} і складової зворотного струму колектора:

$$I_B = I_{BR} - I_{KB0} = (1 - \alpha)I_E - I_{KB0}. \quad (1.22)$$

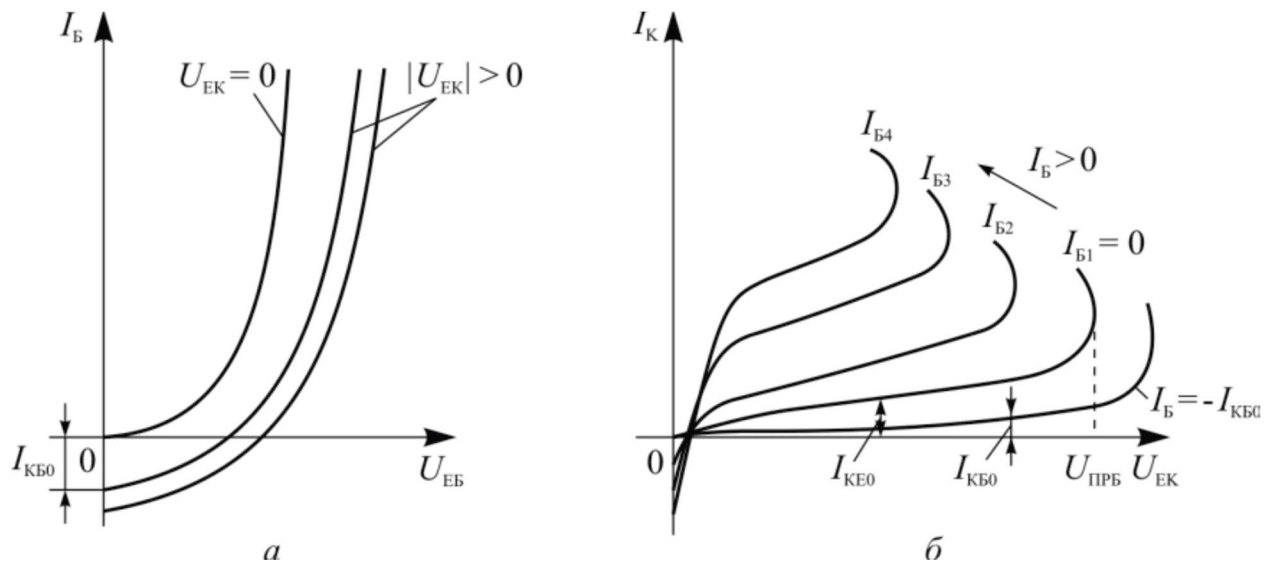


Рисунок 1.7 – Вхідні (а) та вихідні (б) вольт-амперні характеристики транзистора у схемі із загальним емітером [1]

При напрузі на колекторі, що дорівнює нулю, в колі бази тече виключно рекомбінаційний струм і ВАХ приймає вигляд експоненти:

$$I_B = (1 - \alpha) I_{ЕБ0} \left(e^{\frac{U_{ЕБ}}{\varphi_T}} - 1 \right). \quad (1.23)$$

Коли на колекторний перехід подається зворотна напруга, від зазначеної складової віднімається струм $I_{КБ0}$. Тому зі збільшенням напруги $|U_{БК}|$ вхідні характеристики мають зсув праворуч і перетинають вісь напруги.

В свою чергу, вихідні характеристики у схемі із загальним емітером відображають залежність між струмом колектора та напругою на колекторі при постійному струмі бази:

$$I_K = f(U_{ЕК})|_{I_{Б} = \text{const}}. \quad (1.24)$$

Якщо підставити у праву частину рівняння співвідношення $I_E = I_K + I_B$ і виразити струм колектора через струм бази, то вираз (1.24) можна переписати у вигляді:

$$I_K = \frac{\alpha}{1-\alpha} I_B - \frac{I_{KB0}}{1-\alpha} \left(e^{\frac{U_{ЕК}}{\varphi_T}} - 1 \right) = \beta I_B - I_{KE0} \left(e^{\frac{U_{ЕК}}{\varphi_T}} - 1 \right). \quad (1.25)$$

де β - коефіцієнт підсилення за струмом у схемі із загальним емітером; I_{KE0} - зворотний колекторний струм (струм у колі колектора при умові, що струм бази дорівнює нулю) у схемі із загальним емітером.

На рисунку 1.7 (б) можна побачити характер сім'ї вихідних характеристик. Через те, що струм I_{KE0} більший за струм I_{KB0} у схемі із загальною базою, характер дещо інший, бо через колекторний перехід протікає струм I_{KB0} і струм дірок, які інжектуються з емітера в базу.

З урахуванням того, що $I_E = I_K + I_B$, запишемо співвідношення між струмами I_{KE0} і I_{KB0} :

$$I_K = \alpha I_K + \alpha I_B + I_{KB0}. \quad (1.26)$$

Струм колектора I_K , коли струм бази дорівнює нулю, відповідає струму I_{KE0} , тобто:

$$I_K|_{I_B=0} = I_{KE0} = \frac{I_{KB0}}{1-\alpha} = (\beta + 1)I_{KB0}. \quad (1.27)$$

Якщо струми бази не дорівнюють нулю, то вихідні характеристики перетинають вісь абсцис на початкових відрізках. У випадку, коли напруга на колекторі дорівнює нулю (режим короткого замикання на виході), струм колектора I_K змінює напрямок (рис. 1.7, б). Підвищення струму I_B призводить до зміщення характеристик вгору. Відстань між сусідніми характеристиками при однакових приростах струму бази значно варіюється, на відміну від схеми із загальною базою, де ці зміни більш передбачувані. Характеристики мають більший і різноманітний нахил до осі напруг, що ускладнює аналіз та прогнозування поведінки транзистора в різних режимах.

Ці відмінності пояснюються більш чутливою залежністю коефіцієнта β від вхідного струму та вихідної напруги, що значно відрізняється від менш вираженої залежності коефіцієнта α у схемі із загальною базою. Крім того, такі особливості вихідних характеристик можуть впливати на стабільність роботи транзистора, особливо в динамічних режимах, де важливо враховувати нелінійні спотворення та зміни параметрів під навантаженням [5].

Характеристики прямої передачі у схемі із загальним емітером виражають залежність струму колектора від струму бази при постійній напрузі на колекторі:

$$I_K = f(I_E)|_{U_{EK} = \text{const}} \quad (1.28)$$

В рівнянні (1.25) можна побачити, що при умові $|U_{EK}| \gg \varphi_T$, зв'язок між струмом колектора та струмом бази можна описати наступним виразом:

$$I_K = \beta I_B + I_{KE0} \quad (1.29)$$

Через більшу залежність коефіцієнта β , порівняно з коефіцієнтом α , від режиму роботи транзистора характеристики прямої передачі у схемі із загальним емітером відрізняються більшою нелінійністю (рис. 1.8, а).

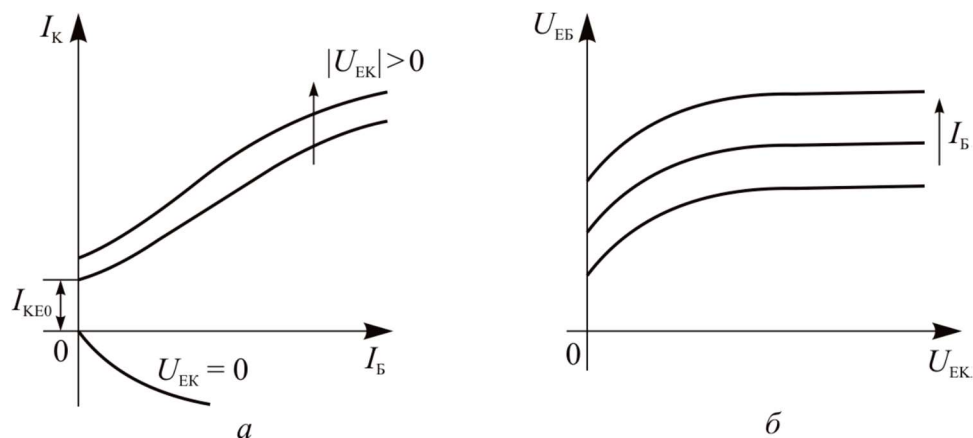


Рисунок 1.8 - Вольт – амперні характеристики прямої передачі (а) та зворотного зв'язку (б) транзистора у схемі із загальним емітером [1]

Залежність напруги на емітерному переході від напруги на колекторі при постійному струмі бази є характеристиками зворотного зв'язку у схемі із загальним емітером:

$$U_{EB} = f(U_{EK})|_{I_B = \text{const}}. \quad (1.30)$$

Оскільки напруга U_{EK} у схемі зі загальним емітером прикладена до обох переходів транзистора, то з підвищенням U_{EK} буде зростати і напруга U_{EB} (рис. 1.8, б), а з підвищенням струму бази характеристики зворотного зв'язку зміщуються вгору, оскільки при підвищенні струму бази збільшується прямий спад напруги на емітерному переході.

1.3 Математичні моделі біполярного транзистора

Біполярний транзистор є одним з найпоширеніших компонентів сучасних електронних пристроїв. За останні десятиліття було розроблено декілька математичних моделей, що описують характеристики біполярного транзистора. Математична модель транзистора представляє собою комбінацію еквівалентної схеми та математичних виразів, що описують її елементи.

Серед численних моделей, створених для аналізу біполярних транзисторів, найбільш відомими є моделі Еберса-Молла, Логана, ПАЕС, Лінвіла, Гумеля-Пуна, IBIS тощо. Відмінності між цими моделями зумовлені різними фізичними ефектами, які враховуються, а також застосованими наближеннями при аналізі різних режимів роботи транзистора.

1.3.1 Математична модель Еберса-Молла

Модель Еберса-Молла, створена Джоном Еберсом та Джоном Моллом у 1954 році, є однією з найфундаментальніших моделей для аналізу біполярних транзисторів. Вона використовується для опису поведінки транзистора в якості підсилювача або перемикача як у статичних, так і динамічних режимах. Модель враховує впливи ключових параметрів та процесів, що відбуваються в транзисторі, і дозволяє створювати точні математичні моделі для аналізу електронних схем. Завдяки розвитку обчислювальної техніки та методів аналізу електронних схем ця модель стала основою для багатьох програмних пакетів, таких як Micro-Cap, Design Center та SPICE.

Основні принципи моделі Еберса-Молла полягають у використанні еквівалентної схеми, що включає два джерела струму та два р-п переходи, які моделюють поведінку емітера та колектора транзистора. Модель враховує як прямі, так і зворотні процеси, що відбуваються в транзисторі, дозволяючи аналізувати його поведінку в різних режимах роботи.

Модель Еберса-Молла описується системою нелінійних рівнянь, що визначають залежності між струмами і напругами на переходах транзистора. Для п-р-п-транзистора рівняння мають вигляд [6]:

$$I_E = \frac{I_{e0}}{1 - \alpha_N \alpha_I} \left(e^{\frac{U_E}{\varphi_0}} - 1 \right) - \frac{\alpha_I I_{k0}}{1 - \alpha_N \alpha_I} \left(e^{\frac{U_K}{\varphi_0}} - 1 \right), \quad (1.31)$$

$$I_K = \frac{\alpha_N I_{e0}}{1 - \alpha_N \alpha_I} \left(e^{\frac{U_E}{\varphi_0}} - 1 \right) - \frac{I_{k0}}{1 - \alpha_N \alpha_I} \left(e^{\frac{U_K}{\varphi_0}} - 1 \right), \quad (1.32)$$

$$I_B = \frac{(1 - \alpha_N) I_{e0}}{1 - \alpha_N \alpha_I} \left(e^{\frac{U_E}{\varphi_0}} - 1 \right) - \frac{(1 - \alpha_I) I_{k0}}{1 - \alpha_N \alpha_I} \left(e^{\frac{U_K}{\varphi_0}} - 1 \right), \quad (1.33)$$

де:

- I_{e0}, I_{k0} — струми насичення для емітера і колектора;

- α_N, α_I — нормальний та інверсний коефіцієнти підсилення струму;
- U_{e0}, U_{k0} — напруги на переходах база-емітер і база-колектор;
- ϕ_0 — тепловий потенціал.

Рівняння (1.31-1.33) застосовують для аналізу статичних режимів транзистора при будь-яких комбінаціях струмів на переходах.

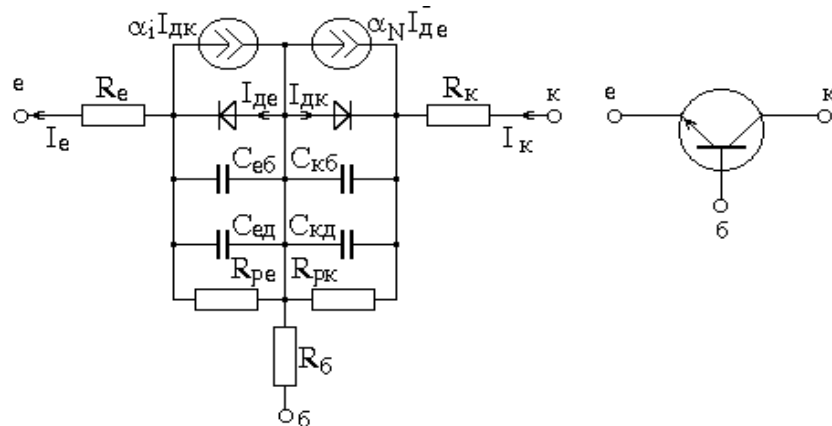


Рисунок 1.8 - Еквівалентна схема моделі Еберса-Молла для n-p-n транзистора (а) та графічне зображення цієї схеми (б) [7]

Еквівалентна схема моделі Еберса-Молла для n-p-n-транзистора зображена на рис. 1.8, а та включає в себе:

- Діоди, що моделюють p-n переходи емітера і колектора. Діоди V_{DE} та V_{DK} відповідають за передачу струму через емітерний та колекторний переходи відповідно.
- Керовані джерела струму, які враховують підсилення струму між емітером, базою та колектором. Наприклад, джерело струму $J = \alpha_F \times I_{DE}$ моделює передачу струму з емітера до колектора.
- Паразитні ємності, які моделюють внутрішні ємнісні ефекти в транзисторі, такі як бар'єрні ємності C_{BE} та C_{BC} , а також дифузійні ємності C_{DE} та C_{DK} .
- Опори, що враховують резистивні властивості базової, колекторної та емітерної областей, наприклад, резистор R_b , що відображає опір базової області транзистора.

Для аналізу роботи транзисторів у режимі малих сигналів модель Еберса-Молла може бути спрощена. Це дозволяє отримати малосигнальні еквівалентні схеми, де нелінійні елементи замінюються лінійними, значення параметрів яких відповідають диференційним значенням у статичній робочій точці. Ці схеми використовуються для аналізу підсилювачів, фільтрів та інших лінійних електронних пристроїв. Наприклад, у малосигнальній еквівалентній схемі діоди V_{DE} та V_{DK} замінюються диференційними опорами R_{BE} та R_{BC} , а бар'єрні ємності C_{BE} та C_{BC} — диференційними ємностями.

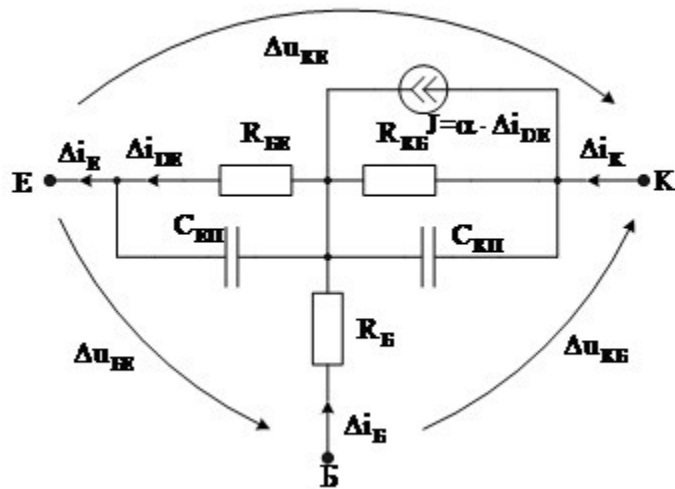


Рисунок 1.9 - Еквівалентна малосигнальна схема біполярного транзистора [12]

Модель Еберса-Молла також дозволяє враховувати частотні властивості транзистора, які проявляються в діапазоні високих частот (сотні кГц і вище). Врахування паразитних ємностей C_{BE} та C_{BC} у малосигнальній еквівалентній схемі дозволяє точно моделювати частотні характеристики транзистора. Значення параметрів елементів еквівалентної схеми можуть варіюватися: R_B та R_{BE} — від десятків Ом до одиниць кОм; R_{BC} — від десятків до сотень кОм; $\alpha=0.95-0.995$; C_{BE} — від сотень до тисяч пФ; C_{BC} — від одиниць до сотень пФ.

1.3.2 Математична модель Гуммеля-Пуна

Модель Гуммеля-Пуна є однією з ключових моделей для аналізу біполярних транзисторів, особливо в умовах високої точності. Розроблена на основі моделі Еберса-Молла. Ця модель враховує додаткові ефекти другого порядку, такі як рекомбінація в області об'ємного заряду, зниження коефіцієнта підсилення по струму при великих токах та ефект Ерлі. Ці особливості роблять модель Гуммеля-Пуна більш придатною для використання в сучасних програмних пакетах для аналізу електронних схем, таких як SPICE [7].

Модель Гуммеля-Пуна зображена на рис. 1.10 та складається з наступних ключових компонентів:

- Ідеальні діоди, що моделюють р-п переходи.
- Ідеальні керовані джерела струму, що моделюють передачу струму через транзистор.
- Паразитні ємності, що враховують ефекти внутрішніх ємностей транзистора.
- Опори, що враховують резистивні властивості базової, колекторної та емітерної областей.

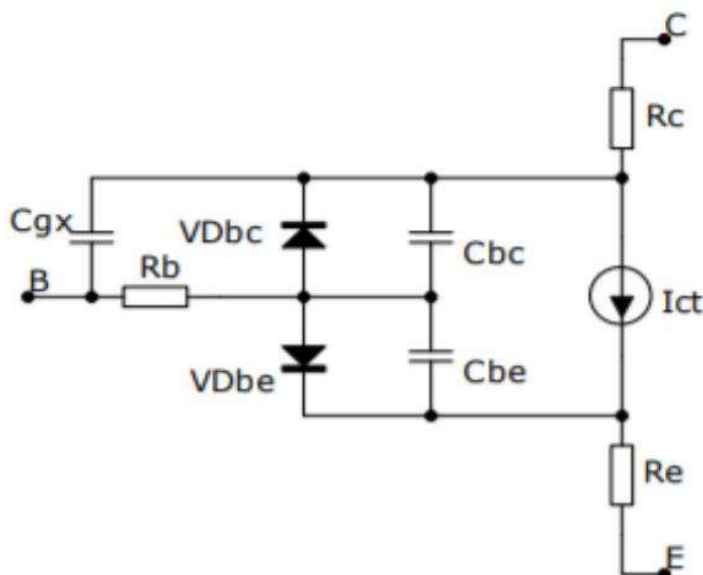


Рисунок 1.10 - Схема заміщення моделі Гуммеля-Пуна переходів база-емітер і база-колектор. Дана модель дозволяє описувати як пряме, так і інверсне включення біполярного транзистора [7]

Модель Гуммеля-Пуна використовує модифіковані рівняння моделі Еберса-Молла для врахування ефектів другого порядку. Рівняння для n-p-n-транзистора має вигляд:

$$J_n = J_s \left(\exp\left(\frac{qV_{BC}}{kT}\right) - \exp\left(\frac{qV_{BE}}{kT}\right) \right). \quad (1.34)$$

Щільність струму зв'язку J_n виражається через повний струм I_n :

$$I_n = I_s \left(\exp\left(\frac{V_{BC}}{V_T}\right) - \exp\left(\frac{V_{BE}}{V_T}\right) \right). \quad (1.35)$$

Модель Гуммеля-Пуна включає такі рівняння:

$$I_C = -I_n - I_s B_r \left(\exp\left(\frac{V_{BC}}{V_T}\right) - 1 \right), \quad (1.36)$$

$$I_E = I_n - I_s B_f \left(\exp\left(\frac{V_{BE}}{V_T}\right) - 1 \right), \quad (1.37)$$

$$I_C = I_s B_f \left(\exp\left(\frac{V_{BE}}{V_T}\right) - 1 \right) + I_s B_r \left(\exp\left(\frac{V_{BC}}{V_T}\right) - 1 \right), \quad (1.38)$$

де I_s , B_f , B_r — параметри моделі, що визначають основні співвідношення моделі Еберса-Молла; V_T — тепловий потенціал, q — заряд електрона, k — стала Больцмана, T — температура в кельвінах; V_{BE} та V_{BC} — напруги на переходах база-емітер та база-колектор.

Модель Гуммеля-Пуна враховує три ключові ефекти другого порядку:

- 1. Рекомбінація в області об'ємного заряду емітерного переходу:** враховує рівняння діодів, що описують струми р-n переходів, додаючи параметри, які дозволяють визначати базовий струм як суперпозицію струмів ідеального і неідеального діода.

2. Зниження коефіцієнта підсилення по струму при великих токах:

Описує зниження підсилення через ефекти високого рівня інжекції.

3. Ефект Ерлі: Вплив розширення області об'ємного заряду на струм зв'язку між емітером і колектором.

Ця модель враховує інжекційні процеси, що дозволяють описати електричні процеси як у статичних, так і в динамічних режимах роботи транзистора. Врахування рекомбінації в області об'ємного заряду та високого рівня інжекції дозволяє отримати більш точне моделювання транзисторів в різних умовах.

Перевагами моделі Гуммеля-Пуна являються висока точність моделювання завдяки врахуванню ефектів другого порядку, а також зручність для автоматизованого моделювання складних електронних схем. А обмеженнями моделі є складність в реалізації через велику кількість параметрів та можливість помилок при високих частотах та екстремальних режимах роботи транзистора.

1.3.3 Математична модель переносу

Модель переносу, іноді відома як модель Логана, є ще однією з математичних моделей біполярного транзистора і важлива для аналізу біполярних транзисторів. Ця модель описує процеси, що відбуваються в транзисторі, виходячи з струмів, які переносяться через базову область від колектора до емітера. Вона враховує особливості, які не завжди враховані в інших моделях, таких як модель Еберса-Молла, і дозволяє детальніше аналізувати поведінку транзистора в різних режимах роботи.

Модель переносу базується на описі струмів, що переносяться через базову область транзистора, та враховує вплив напруги на емітерному та колекторному

переходах. Вона також включає компоненти, що відображають різні ефекти, такі як ємності переходів і опори областей транзистора.

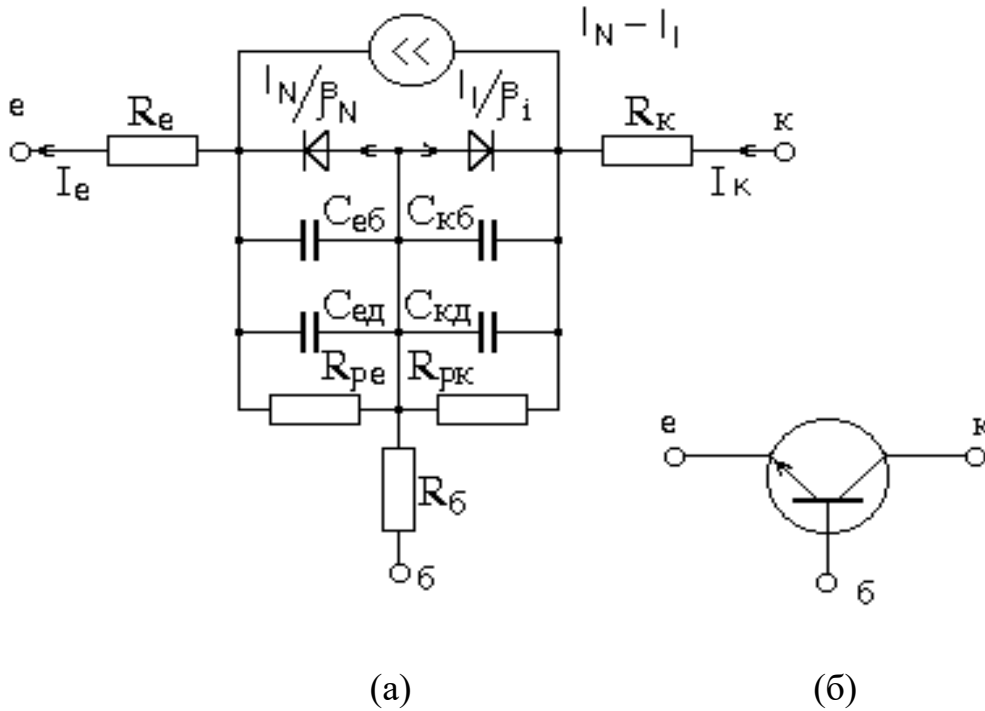


Рисунок 1.11 - Еквівалентна схема моделі переносу для p–p–n транзистора (а) і графічне зображення даної схеми (б) [13]

Система рівнянь моделі переносу описує взаємозв'язок струмів та напруг у транзисторі. Для p–p–n-транзистора ці рівняння мають вигляд [13]:

$$I_E = \left(1 + \frac{1}{\beta_N}\right) \cdot I_N - I_I + C_E \frac{dU}{dt} + \frac{U_E}{R_{BE}}, \quad (1.39)$$

$$I_K = \left(1 + \frac{1}{\beta_I}\right) \cdot I_I - I_N + C_K \frac{dU}{dt} + \frac{U_K}{R_{BK}}, \quad (1.40)$$

$$I_N = I_S \left(\exp \frac{U_e}{\phi_T} - 1 \right); I_I = I_S \left(\exp \frac{U_K}{\phi_T} - 1 \right), \quad (1.41)$$

$$C_E = C_{Eb} + C_{Ed}; C_K = C_{Kb} + C_{Kd}; \quad (1.42)$$

$$C_{EL} = \frac{\tau_N}{\phi_T} (I_N + I_S); C_{KL} = \frac{\tau_I}{\phi_T} (I_I + I_S), \quad (1.43)$$

де I_E, I_K - струми емітера та колектора; I_N, I_I - струми переносу, які збираються переходами в прямому та інверсному включенні транзистора; β_N, β_I - нормальний та інверсний коефіцієнти підсилення по струму в схемі із спільним емітером; U_E, U_K - напруга на емітерному та колекторному переходах; I_S - тепловий струм переходу; C_E, C_{EB}, C_{UD} – сумарна, бар’єрна та дифузійна ємності емітерного переходу; C_K, C_{KB}, C_{KD} – сумарна, бар’єрна та дифузійна ємності колекторного переходу. φ_t - тепловий потенціал; φ_E, φ_K - контактна різниця потенціалів переходів; τ_N, τ_I - сталі часу накопичування зарядів.

Еквівалентна схема моделі переносу зображена на рисунку 1.11 і включає:

- Діоди, що моделюють р-п переходи емітера і колектора.
- Генератори струму, що відображають передачу струму через базову область транзистора.
- Паразитні ємності, що враховують внутрішні ємнісні ефекти в транзисторі.
- Опори, що враховують резистивні властивості базової, колекторної та емітерної областей.

Перевагами моделі переносу є доволі висока точність моделювання завдяки врахуванню внутрішніх процесів у базовій області транзистора, можливість врахування частотних характеристик транзистора, але обмеженнями моделі є складність в реалізації через велику кількість параметрів та можливість помилок при високих частотах та екстремальних режимах роботи транзистора.

1.4 Висновки

В цьому розділі було розглянуто фундаментальні принципи та характеристики роботи біполярного транзистора, а також було проведено детальний аналіз математичних моделей, що описують поведінку біполярних транзисторів. Детально

проаналізовано вольт-амперні характеристики для схем із загальною базою та загальним емітером, що відображають залежність струмів емітера, бази та колектора від прикладених напруг.

Розділ також містить аналіз математичних моделей біполярного транзистора, таких як модель Еберса-Молла, модель Гуммеля-Пуна та модель переносу. Модель Еберса-Молла є основною для аналізу підсилювальних і перемикальних властивостей транзистора, тоді як модель Гуммеля-Пуна враховує ефекти другого порядку, такі як рекомбінація в області об'ємного заряду та зниження коефіцієнта підсилення при великих струмах. Модель переносу зосереджена на струмах, що переносяться через базову область транзистора, і враховує ємності переходів і опори областей транзистора.

Отже, у розділі наведено комплексний аналіз фізичних процесів і характеристик роботи біполярного транзистора, а також різні підходи до моделювання його властивостей в статичних та динамічних режимах.

2 ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИКА ДОСЛІДЖЕННЯ

2.1 Формулювання задачі

У цьому розділі буде визначено конкретні завдання, які необхідно виконати в рамках даної магістерської роботи. Мета дослідження полягає в розробці ефективного методу розрахунку ВАХ біполярного транзистора, за допомогою програмної системи для автоматизованої побудови та аналізу вольт-амперних характеристик (ВАХ) біполярних транзисторів на основі їх математичних моделей, зокрема моделі Еберса-Молла. Система повинна забезпечити можливість введення даних математичної моделі Еберса-Молла для побудови ВАХ, триангуляцію поверхні ВАХ та оптимізацію пошуку в базі даних для швидкого визначення точок.

Основні завдання дослідження

1. Розробка алгоритму побудови тривимірної поверхні ВАХ.

Спроекувати алгоритм для обчислення вольт-амперних характеристик біполярних транзисторів на основі введених користувачем даних. Реалізувати алгоритм побудови поверхні ВАХ із подальшою триангуляцією.

2. Введення користувачем даних для побудови ВАХ.

Розробити інтерфейс для введення користувачем даних моделі Еберса-Молла.

3. Триангуляція утвореної поверхні ВАХ.

Реалізувати алгоритм триангуляції поверхні ВАХ для розбиття її на трикутники із наперед заданою точністю. Забезпечити високу точність триангуляції для подальшого аналізу характеристик.

4. Збереження даних.

Створити структуру бази даних для зберігання точок триангульованої поверхні ВАХ та параметрів транзисторів. Забезпечити надійне зберігання та доступ до даних у базі даних.

5. Оптимізація пошуку в базі даних.

Розробити алгоритми для ефективного пошуку точок у базі даних за двома координатами з метою визначення третьої з наперед заданою точністю. Реалізувати оптимізацію запитів до бази даних для підвищення швидкодії системи.

6. Візуалізація результатів.

Розробити модуль візуалізації для відображення результатів обчислень у зручному для користувача форматі (графіки, таблиці тощо). Забезпечити інтерактивність візуалізації для полегшення аналізу результатів.

7. Тестування та валідація системи.

Провести тестування системи на різних тестових моделях біполярних транзисторів для перевірки її точності та надійності. Валідувати результати обчислень шляхом порівняння з теоретичними даними та експериментальними результатами.

8. Очікувані результати.

Результатом виконання зазначених завдань має стати розроблена програмна система для оптимізованого розрахунку біполярних транзисторів, яка забезпечить автоматизоване обчислення та аналіз ВАХ транзисторів з високою швидкістю. Система буде корисним інструментом для інженерів та науковців у галузі електроніки, дозволяючи швидко і ефективно визначати та застосовувати лінеаризовані характеристики транзисторів.

2.2 Методи триангуляції в комп'ютерній графіці та чисельних розрахунках

Триангуляція — це процес розбиття геометричних об'єктів на тривіальні елементи, такі як трикутники, для спрощення подальшого аналізу, обчислень і рендерингу. У комп'ютерній графіці та чисельних розрахунках триангуляція є важливим інструментом, що дозволяє вирішувати складні задачі, такі як обчислення площ, об'ємів, інтерполяція функцій і моделювання фізичних процесів. Кожен трикутник визначається трьома вершинами (точками), і такі трикутники не повинні перекриватися, але можуть ділити спільні ребра та вершини.

Основні властивості триангуляції включають суміжність, повноту та неперервність. Властивість суміжності полягає в тому, що трикутники можуть мати спільні ребра та вершини, але не повинні перекриватися. Повнота - область, яка піддається триангуляції, повністю покривається трикутниками без прогалин. Неперервність - межі трикутників утворюють безперервну структуру без розривів.

Існує кілька основних методів триангуляції, кожен з яких має свої особливості та застосування, такі як триангуляція Делоне, алгоритм Рупперта, алгоритм Боулера-Вотсона, Grid-Based триангуляція, Edge Subdivision, Midpoint Subdivision і Centroid Subdivision.

2.2.1 Триангуляція Делоне

Триангуляція Делоне є одним із найпопулярніших методів триангуляції завдяки своїм корисним властивостям у чисельних методах і геометричному моделюванні [14]. Основні властивості Делоне включають:

- Максимально мінімальний кут. Це означає, що найменший кут серед усіх кутів трикутників у триангуляції Делоне є максимально можливим. Це допомагає уникати "тонких" трикутників, які можуть бути проблематичними для чисельної стабільності і точності розрахунків;

- Відсутність точок всередині описаного кола. Жодна точка не повинна знаходитися всередині описаного кола будь-якого трикутника в триангуляції Делоне. Це забезпечує властивість Делоне, яка є ключовою для побудови стійких сіток.

Основний алгоритм триангуляції Делоне полягає в наступному [15]:

1. Початкова розбивка - область розбивається на великі початкові трикутники, які охоплюють всі точки.

2. Додавання нових точок - нові точки додаються при триангуляції послідовно.

3. Перебудова трикутників - для збереження властивостей Делоне використовуються операції «фліпування» ребер. «Фліпування» включає зміну діагоналі чотирикутника, утвореного двома трикутниками, для забезпечення відповідності критеріям Делоне.

Цей метод широко використовується в комп'ютерній графіці для побудови високоякісних сіток, а також у чисельних методах, таких як метод кінцевих елементів.

2.2.2 Алгоритм Рупперта

Алгоритм Рупперта спрямований на створення якісної триангуляції з додатковими вимогами до якості трикутників, такими як мінімальний кут [16].

Основні етапи алгоритму включають:

1. Початкова триангуляція - спочатку створюється груба триангуляція області.
2. Перевірка якості трикутників - кожен трикутник перевіряється на відповідність критеріям якості, зокрема кутам.
3. Додавання нових точок - у трикутники, які не задовольняють критеріям якості, додаються нові точки, зазвичай у центр описаного кола або у центр найменшого кута.
4. Перебудова трикутників - триангуляція перебудовується для забезпечення відповідності новим точкам і збереження якості.

Цей метод є корисним у задачах, де важлива висока якість трикутників, наприклад, у чисельних розрахунках для підвищення точності результатів.

2.2.3 Алгоритм Боулера-Вотсона

Алгоритм Боулера-Вотсона розроблений для швидкої генерації триангуляції, де швидкість є пріоритетом над якістю [16]. Основні етапи включають:

1. Початкова велика триангуляція - спочатку створюється велика триангуляція, яка охоплює всі точки.
2. Додавання нових точок - нові точки додаються випадковим чином або відповідно до заданого розподілу.
3. Уточнення триангуляції - після додавання нових точок триангуляція уточнюється шляхом перебудови трикутників.

Алгоритм використовується в додатках, де потрібна швидка генерація сіток, наприклад, у реальному часі або для попереднього аналізу даних.

2.2.4 Grid-Based триангуляція

Grid-Based триангуляція використовує регулярну сітку для розміщення точок і розбиття області на трикутники [17]. Основні етапи та правила включають:

1. Розміщення точок на регулярній сітці - область розбивається на регулярну сітку квадратів. Точки розміщуються на перетинах сітки або всередині кожного квадрата за певними правилами, такими як центр квадрата або випадкове розміщення.
2. Об'єднання точок у трикутники - після розміщення точок вони об'єднуються у трикутники за певними правилами. Найпоширенішим методом є розбиття кожного квадрата на два трикутники шляхом проведення діагоналі.

Існують два основних варіанти:

- Основний діагональний метод: У кожному квадраті проводиться діагональ від верхнього лівого кута до нижнього правого кута.

- Альтернативний діагональний метод: У кожному квадраті проводиться діагональ від верхнього правого кута до нижнього лівого кута.

Цей метод є зручним для задач, де геометрія області має регулярну структуру або потребує швидкого розбиття, наприклад, у задачах моделювання фізичних процесів на регулярних сітках.

2.2.5 Edge Subdivision

Edge Subdivision — це метод триангуляції, що включає розділення ребер існуючих трикутників для досягнення більшої точності та рівномірності сітки [19].

Основні етапи включають:

1. Вибір ребра для розділення - визначаються ребра, які потребують розділу для покращення сітки.
2. Додавання нової точки на середину ребра - нова точка додається в середину обраного ребра.
3. Перебудова трикутників - навколо нового ребра будуються трикутники, що утворюються новими вершинами. Утворюються нові трикутники, які задовольняють заданим критеріям.

Цей метод використовується для уточнення трикутної сітки, покращення рівномірності та зменшення розміру трикутників, що важливо для чисельних методів.

2.2.6 Midpoint Subdivision

Midpoint Subdivision — метод, що передбачає підрозділ кожного ребра трикутника на дві рівні частини для створення нових точок і трикутників [19].

Основні етапи включають:

1. Визначення середини кожного ребра трикутника - обчислюються середини кожного ребра.
2. Додавання нових точок у середини ребер - нові точки додаються у середини ребер.
3. Розбиття кожного трикутника на чотири нових трикутників - використовуючи нові точки, кожний трикутник розбивається на чотири нових трикутників.

Цей метод застосовується для створення більш гладких і детальних поверхонь, особливо у випадках, коли потрібно покращити точність моделювання, наприклад, у моделюванні поверхонь або рендерингу [20].

2.2.7 Centroid Subdivision

Centroid Subdivision — метод, що передбачає підрозділ трикутника шляхом додавання нової точки у центр трикутника (центроїд) та утворення нових трикутників [19]. Основні етапи включають:

1. Визначення центроїду трикутника - обчислюється центроїд кожного трикутника.
2. Додавання нової точки у центроїд - нова точка додається у центроїд трикутника.
3. Розбиття трикутника на три нові трикутники - використовуючи нову точку, кожний трикутник розбивається на три нові трикутники.

Цей метод використовується для створення більш рівномірних сіток і зменшення розміру трикутників для покращення точності моделювання, особливо у чисельних методах і комп'ютерній графіці.

2.2.8 Застосування триангуляції

Комп'ютерна графіка:

- Моделювання поверхонь. Триангуляція використовується для побудови тривимірних моделей об'єктів шляхом розбиття поверхонь на трикутники. Це дозволяє зручно обробляти та візуалізувати складні форми.

- Рендеринг. У процесі рендерингу зображень трикутники використовуються як основні примітиви для побудови сцен. Вони легко обробляються графічними процесорами (GPU), що забезпечує швидке відображення.

- Анімація. Триангуляція дозволяє створювати сітки для моделювання деформацій та рухів об'єктів, що є критичним для реалістичної анімації.

Чисельні розрахунки:

- Метод кінцевих елементів (МКЕ) [20]. Використовує триангуляцію для розбиття складних геометричних областей на простіші елементи (трикутники або тетраедри), що дозволяє ефективно вирішувати диференціальні рівняння.

- Інтерполяція та апроксимація. Триангуляція застосовується для побудови сіток, на яких здійснюється інтерполяція значень функцій, що є важливим для моделювання фізичних процесів.

- Геостатистика. Використовує триангуляцію для побудови моделей просторових даних та проведення інтерполяцій, що є важливим для аналізу геологічних та екологічних даних.

2.3 Програмні засоби і середовища розробки

У цьому пункті буде розглянуто вибір програмних засобів та середовищ розробки для реалізації програмної системи, призначеної для аналізу вольт-амперних

характеристик (ВАХ) біполярних транзисторів. Враховуючи необхідність обробки великого обсягу даних та виконання складних обчислень, мова програмування повинна бути ефективною, потужною та забезпечувати високу продуктивність.

2.3.1 Мови програмування

Сучасні мови програмування пропонують різноманітні інструменти для вирішення математичних задач [21]. У цьому контексті ми розглянемо найбільш відповідні мови програмування для математичних розрахунків та обґрунтуємо їх вибір для аналізу ВАХ біполярних транзисторів.

1. Python

Python є однією з найпопулярніших мов програмування завдяки своїй універсальності та широкому спектру бібліотек, орієнтованих на наукові обчислення. Зокрема, бібліотеки NumPy і SciPy забезпечують високу продуктивність для обробки великих масивів даних та реалізації складних математичних операцій [25]. Pandas пропонує зручний інструментарій для маніпуляцій з табличними даними, що є ключовим у задачах аналізу ВАХ.

Python також має потужний модуль для виконання математичних операцій, таких як експоненціальні, логарифмічні та тригонометричні функції. Використання Python дозволяє легко інтегрувати ці математичні обчислення у програмну систему завдяки його зрозумілому синтаксису та наявності багатьох готових рішень для візуалізації результатів [24]. Окрім того, Python підтримує паралельні обчислення, що може значно прискорити обробку великих обсягів даних.

2. R

R є спеціалізованою мовою програмування, орієнтованою на статистичний аналіз та візуалізацію даних. Її використання є виправданим у контексті обробки та

аналізу експериментальних даних ВАХ, оскільки вона має потужний інструментарій для статистичних обчислень і побудови графіків.

Бібліотеки, такі як `ggplot2`, надають можливості для створення високоякісних графічних зображень, що дозволяє наочно представляти результати аналізу. Інструменти для статистичних тестів, включаючи *t*-тести та регресійний аналіз, дозволяють глибоко аналізувати отримані дані та робити науково обґрунтовані висновки.

3. MATLAB

MATLAB є потужним середовищем для чисельних обчислень, широко використовуваним у наукових дослідженнях та інженерних розробках. Його можливості включають аналіз даних, розробку алгоритмів, моделювання та симуляцію фізичних процесів [33]. MATLAB має вбудовані функції для роботи з матрицями, розв'язування систем лінійних рівнянь, чисельної інтеграції та диференціювання.

Особливо корисним для задач аналізу ВАХ є інструментарій MATLAB для обробки сигналів і зображень, що дозволяє аналізувати і візуалізувати дані вимірювань. MATLAB також підтримує паралельні обчислення та може бути інтегрований з іншими мовами програмування, такими як C/C++ та Python, що розширює його можливості.

4. Wolfram Mathematica

Wolfram Mathematica є високопродуктивним програмним середовищем для виконання символічних і чисельних обчислень. Її використання виправдане у випадках, коли потрібна висока точність і ефективність обчислень, а також гнучкість у створенні математичних моделей.

Mathematica забезпечує широкий спектр алгоритмів для різних математичних задач, включаючи розв'язування диференціальних рівнянь, оптимізацію, статистичний аналіз та візуалізацію даних. Її інтеграція з іншими мовами

програмування та підтримка високопродуктивних обчислень роблять її потужним інструментом для наукових досліджень.

5. Fortran

Fortran, будучи однією з найстаріших мов програмування, залишається актуальною завдяки своїй ефективності у чисельних обчисленнях. Вона широко використовується у наукових та інженерних дисциплінах для моделювання фізичних процесів, розв'язування систем диференціальних рівнянь та обробки великих обсягів даних.

Fortran є особливо корисним у задачах, що вимагають високої точності та ефективності обчислень. Її оптимізовані компілятори забезпечують швидке виконання програм, що важливо для аналізу великих наборів даних, таких як ВАХ біполярних транзисторів.

6. SAS

SAS є потужним інструментом для статистичного аналізу та обробки даних, широко використовуваним у бізнес-аналітиці та наукових дослідженнях. Його можливості включають аналіз великих наборів даних, побудову моделей, виявлення закономірностей та прогнозування.

У контексті аналізу ВАХ SAS може бути використаний для статистичного аналізу експериментальних даних, виявлення трендів та аномалій, а також побудови прогнозних моделей. Його інтеграція з базами даних та іншими програмними засобами робить його потужним інструментом для комплексного аналізу.

7. Julia

Julia є відносно новою мовою програмування, орієнтованою на високопродуктивні чисельні обчислення. Вона поєднує простоту використання з продуктивністю, що робить її особливо привабливою для наукових досліджень та інженерних розробок. Julia має синтаксис, що дозволяє легко писати математичні вирази, і підтримує багатопотокові та паралельні обчислення, що робить її

ефективною для обробки великих обсягів даних. Вона також інтегрується з іншими мовами програмування та підтримує бібліотеки для чисельних обчислень, машинного навчання та візуалізації даних.

8. Maple

Maple є спеціалізованим середовищем для математичних обчислень, що забезпечує широкий спектр інструментів для розв'язування складних математичних задач. Вона використовується в освіті, наукових дослідженнях та інженерних розробках завдяки своїм потужним символічним і чисельним обчисленням.

Maple забезпечує зручний інтерфейс для написання математичних виразів і включає велику бібліотеку функцій для обчислень у галузях, таких як чисельний аналіз, теорія чисел, комбінаторика та лінійна алгебра. Її інструменти для візуалізації даних дозволяють наочно представляти результати обчислень, що є важливим у наукових дослідженнях.

Вибір мови програмування для реалізації програмної системи, призначеної для аналізу вольт-амперних характеристик біполярних транзисторів, залежить від вимог до продуктивності, точності обчислень та зручності використання. Python, MATLAB, Wolfram Mathematica, R, Fortran, SAS, Julia та Maple пропонують різні підходи та інструменти для розв'язання складних математичних задач і можуть бути ефективно використані залежно від конкретних вимог проекту.

2.3.2 Бази даних

Бази даних відіграють важливу роль у зберіганні, управлінні та обробці великих обсягів даних, що виникають у процесі дослідження вольт-амперних характеристик (ВАХ) біполярних транзисторів. Вибір відповідної системи управління базами даних

(СУБД) є важливим для забезпечення надійності, ефективності та масштабованості програмної системи. У цьому розділі буде розглянуто основні типи баз даних та їх застосування.

Реляційні бази даних (SQL)

Реляційні бази даних (РБД) є найпоширенішим типом СУБД, який використовує структуровану мову запитів (SQL) для маніпуляції даними [26].

Основні переваги РБД включають:

- Структурованість даних: дані організовані у вигляді таблиць, що дозволяє легко моделювати складні взаємозв'язки між різними типами даних.
- Надійність та цілісність: підтримка транзакцій та механізмів збереження цілісності даних гарантує точність та консистентність інформації.
- Масштабованість: сучасні РБД підтримують горизонтальне та вертикальне масштабування, що дозволяє обробляти великі обсяги даних.

Прикладами реляційних баз даних є PostgreSQL (відкритий код, потужна реляційна СУБД, яка підтримує розширені функції, такі як збережені процедури, тригери, і складні запити), MySQL (широко використовувана СУБД з відкритим кодом, відома своєю продуктивністю та надійністю. Підтримує різні типи індексів та транзакції), Microsoft SQL Server (комерційна СУБД від Microsoft, яка пропонує високий рівень інтеграції з іншими продуктами Microsoft, забезпечуючи надійність та масштабованість)

NoSQL бази даних

NoSQL бази даних забезпечують більшу гнучкість у порівнянні з реляційними, що робить їх придатними для роботи з неструктурованими або слабо структурованими даними [26][27].

Основні типи NoSQL баз даних включають:

- Документні бази даних: зберігають дані у форматі документів (наприклад, JSON), що дозволяє зберігати складні структури даних.
- Колонкові бази даних: орієнтовані на зберігання великих обсягів даних, організованих по стовпцях, що забезпечує високу продуктивність при виконанні аналітичних запитів.
- Графові бази даних: використовуються для моделювання складних взаємозв'язків між даними.

Прикладами NoSQL баз даних є MongoDB (документно-орієнтована СУБД, яка забезпечує високу гнучкість та масштабованість, що дозволяє зберігати дані у форматі BSON (бінарний JSON), що полегшує маніпулювання складними даними), Apache Cassandra (колонково-орієнтована СУБД, розроблена для обробки великих обсягів даних з високою доступністю і відмовостійкістю, що використовується у великих розподілених системах), Neo4j (графова СУБД, яка спеціалізується на зберіганні та обробці даних, організованих у вигляді графів, що використовується для моделювання складних взаємозв'язків між об'єктами).

Бази даних у пам'яті (In-Memory Databases)

Бази даних у пам'яті зберігають дані в оперативній пам'яті, що забезпечує дуже високу швидкість доступу та обробки даних [26].

Основні переваги:

- Швидкодія: обробка даних відбувається набагато швидше порівняно з традиційними базами даних, які використовують дискове зберігання.
- Низька затримка: ідеально підходять для додатків, що потребують миттєвого доступу до даних.

Прикладами баз даних у пам'яті є Redis (високопродуктивна база даних, яка працює в оперативній пам'яті і забезпечує миттєвий доступ до даних, що підтримує різні структури даних, такі як рядки, хеші, списки, множини та відсортовані множини), SAP HANA (потужна платформа для обробки даних у пам'яті, яка

забезпечує високу продуктивність для аналітичних і транзакційних додатків, що підтримує обробку великих обсягів даних у режимі реального часу)

Гібридні бази даних

Гібридні бази даних поєднують переваги реляційних та NoSQL баз даних, забезпечуючи як структуроване, так і неструктуроване зберігання даних. Вони підтримують різні моделі даних і дозволяють використовувати різні типи запитів для аналізу[27]. Прикладами гібридних баз даних являються Couchbase (поєднує можливості реляційних та NoSQL баз даних, підтримуючи зберігання документів, ключ-значення та аналітичні запити), Oracle NoSQL Database (поєднує реляційну та NoSQL архітектуру для забезпечення гнучкості та ефективності при обробці великих обсягів даних)

2.3.3 Бібліотеки та інструменти

Для реалізації програмної системи, важливо обрати відповідні бібліотеки та інструменти, які забезпечать ефективне виконання всіх необхідних функцій. У цьому пункті буде розглянуто основні бібліотеки та інструменти, що будуть використані для розробки даної програмної системи.

Вибір бібліотек для обробки та аналізу даних

1. NumPy

NumPy – це модуль з відкритим кодом для Python, який надає загальні математичні та числові операції у вигляді скомпільованих, високошвидкісних функцій. Ці функції об'єднані у високорівневі пакети, що забезпечують функціональність, аналогічну можливостям MatLab. Вона надає підтримку для багатовимірних масивів і матриць, а також велику кількість високорівневих

математичних функцій для роботи з цими масивами. Перевагами даної бібліотеки є швидкість і продуктивність обчислень, а також підтримка векторизованих операцій, що дозволяє значно підвищити ефективність обробки даних. Дана бібліотека має широке застосування в наукових та інженерних дослідженнях [24].

2. SciPy

SciPy розширює функціональність NumPy, надаючи додаткові модулі для оптимізації, інтеграції, інтерполяції, власних значень, спеціальних функцій та інших завдань, що зустрічаються у наукових та інженерних дослідженнях [24].

Вибір бібліотек для візуалізації даних

1. Matplotlib

Matplotlib – це бібліотека для побудови графіків для мови програмування Python та його числового модуля NumPy. Вона надає об'єктно-орієнтований API для вбудовування графіків у додатки, використовуючи універсальні графічні інтерфейси, такі як Tkinter, wxPython, Qt або GTK. Існує також процедурний інтерфейс "pylab", заснований на машині станів (подібно до OpenGL), який призначений для нагадування інтерфейсу MATLAB, хоча його використання не рекомендується. SciPy також використовує Matplotlib [25].

2.4 Опис бази даних та її структури

Для забезпечення ефективного зберігання, управління та обробки даних, пов'язаних із вольт-амперними характеристиками (ВАХ) біполярних транзисторів, необхідно створити структуру бази даних, яка б задовольняла вимоги високої продуктивності, надійності та масштабованості. Вибір відповідної системи

управління базами даних (СУБД) та проектування структури бази даних є критично важливими кроками для досягнення цих цілей.

Основні вимоги до бази даних для зберігання ВАХ біполярних транзисторів включають:

- Збереження великих обсягів даних: ВАХ включають велику кількість точок даних, які повинні бути збережені в базі даних.
- Швидкий доступ до даних: Для аналізу та обробки даних необхідно забезпечити швидкий доступ до потрібної інформації.
- Надійність та цілісність даних: Дані повинні зберігатися надійно та з дотриманням всіх вимог до цілісності.
- Гнучкість та масштабованість: База даних повинна легко масштабуватися та адаптуватися до зміни обсягів даних та вимог системи.

Для досягнення зазначених цілей було обрано реляційну базу даних PostgreSQL, яка задовольняє всім вказаним вимогам. PostgreSQL є потужною, відкритою СУБД, яка підтримує розширені функції для роботи з даними, забезпечує високу продуктивність та надійність.

Структура бази даних розроблена таким чином, щоб забезпечити зручне зберігання та швидкий доступ до даних, пов'язаних з ВАХ біполярних транзисторів. Основні таблиці бази даних включають:

1. Таблиця транзисторів (transistors).
 - id: Унікальний ідентифікатор транзистора (PRIMARY KEY).
 - name: Назва або модель транзистора.
 - type: Тип транзистора (наприклад, NPN або PNP).
 - description: Опис транзистора, включаючи технічні характеристики.

```
CREATE TABLE transistors (  
    id SERIAL PRIMARY KEY,  
    name VARCHAR(100) NOT NULL,  
    type VARCHAR(10),  
    description TEXT
```

);

2. Таблиця вольт-амперних характеристик (vah_data).

- id: Унікальний ідентифікатор точки ВАХ (PRIMARY KEY).
- transistor_id: Ідентифікатор транзистора (FOREIGN KEY, посилання на transistors.id).
- voltage: Значення напруги.
- current: Значення струму.
- measurement_time: Час проведення вимірювання.

```
CREATE TABLE vah_data (
    id SERIAL PRIMARY KEY,
    transistor_id INT REFERENCES transistors(id),
    voltage FLOAT NOT NULL,
    current FLOAT NOT NULL,
    temperature FLOAT,
    measurement_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);
```

3. Таблиця триангуляційних даних (triangulation_data).

- id: Унікальний ідентифікатор трикутника (PRIMARY KEY).
- transistor_id: Ідентифікатор транзистора (FOREIGN KEY, посилання на transistors.id).
- point1_id: Ідентифікатор першої точки трикутника (посилання на vah_data.id).
- point2_id: Ідентифікатор другої точки трикутника (посилання на vah_data.id).
- point3_id: Ідентифікатор третьої точки трикутника (посилання на vah_data.id).


```
CREATE TABLE triangulation_data (
    id SERIAL PRIMARY KEY,
    transistor_id INT REFERENCES transistors(id),
    point1_id INT REFERENCES vah_data(id),
    point2_id INT REFERENCES vah_data(id),
    point3_id INT REFERENCES vah_data(id)
);
```

Для забезпечення надійного зберігання та доступу до даних, база даних PostgreSQL надає наступні можливості.

- Транзакції: Використання транзакцій гарантує цілісність даних та дозволяє виконувати серію операцій як єдину атомарну дію.
- Індeksi: Створення індeксів на полях, що часто використовуються у запитах (наприклад, transistor_id, voltage, current), значно підвищує швидкість доступу до даних.
- Резервне копіювання: Регулярне резервне копіювання бази даних забезпечує можливість відновлення даних у разі непередбачених обставин.
- Моніторинг та оптимізація: Використання інструментів моніторингу та оптимізації запитів (наприклад, pg_stat_statements) дозволяє аналізувати та покращувати продуктивність бази даних.

2.5 Методи оптимізованого пошуку в базах даних

Ефективний пошук даних у базах даних є дуже важливим фактором для забезпечення швидкодії та продуктивності інформаційних систем, особливо у випадках великих обсягів даних. Оптимізація пошуку дозволяє значно зменшити час

доступу до потрібної інформації, що є ключовим для швидкого аналізу та обробки даних.

Індексація є одним з основних методів підвищення швидкодії пошукових операцій у базах даних. Індеси дозволяють швидко знаходити записи в таблиці без необхідності повного сканування. Основні типи індесів включають:

1. B-Tree індеси

B-Tree індеси є найбільш поширеним типом індесів у реляційних базах даних. Вони забезпечують збалансоване дерево, яке дозволяє ефективно виконувати операції пошуку, вставки та видалення з логарифмічною складністю. B-Tree індеси забезпечують швидкий пошук записів при великих обсягах даних, а також має підтримку діапазонних запитів

```
CREATE INDEX idx_voltage ON vah_data(voltage);
CREATE INDEX idx_current ON vah_data(current);
```

2. Hash індеси

Hash індеси використовують хеш-функції для визначення розташування запису. Вони ефективні для точного пошуку за значенням, але не підходять для діапазонних запитів.

```
CREATE INDEX idx_transistor_id ON vah_data USING
hash(transistor_id);
```

3. Геопросторові індеси (R-Tree)

Геопросторові індеси, такі як R-Tree, використовуються для роботи з багатовимірними даними, наприклад, координатами точок у просторі. Вони ефективні для пошуку даних у географічних і багатовимірних додатках. Данна індексація здійснює ефективний пошук за багатовимірними даними, а також має підтримку діапазонних та найближчих сусідів запитів.

```
CREATE EXTENSION postgis;
CREATE INDEX idx_location ON vah_data USING GIST
(ST_MakePoint(voltage, current));
```

4. Кешування запитів дозволяє зберігати результати часто виконуваних запитів у пам'яті, що значно зменшує час їх виконання при повторному зверненні. Це особливо корисно для запитів, що потребують значних обчислювальних ресурсів.

Приклад використання Redis для кешування результатів запитів надано у додатку А.1.

Оптимізація запитів включає в себе покращення структури SQL-запитів для підвищення їх продуктивності. Це може включати:

1. Використання правильних типів JOIN

Оптимізація JOIN операцій може значно вплинути на продуктивність запитів. Використання INNER JOIN замість OUTER JOIN, коли можливо, може зменшити обсяг оброблюваних даних.

2. Обмеження кількості повернутих рядків

Використання оператора LIMIT дозволяє обмежити кількість рядків, які повертає запит, що зменшує навантаження на базу даних та мережу.

```
SELECT * FROM vah_data WHERE transistor_id = 1 LIMIT 100;
```

3. Уникнення підзапитів

У деяких випадках підзапити можуть бути замінені на JOIN операції, що може значно підвищити продуктивність запиту.

-- Низька продуктивність

```
SELECT * FROM vah_data WHERE transistor_id IN (SELECT id
FROM transistors WHERE type = 'NPN');
```

-- Висока продуктивність

```
SELECT vah_data.* FROM vah_data JOIN transistors ON
vah_data.transistor_id = transistors.id WHERE transistors.type
= 'NPN';
```

Також ще одним методом оптимізації пошуку в базах даних є паралельне виконання запитів. Сучасні бази даних, такі як PostgreSQL, підтримують паралельне виконання запитів, що дозволяє розподілити обчислювальне навантаження між декількома процесорами або ядрами. Це може значно підвищити продуктивність при

виконанні складних запитів, покращити продуктивність запитів на багатоядерних системах та зменшити час виконання складних запитів.

2.6 Висновки

В цьому розділі було визначено мету дослідження, яка полягає в розробці програмної системи для ефективного розрахунку ВАХ біполярного транзистора, а також швидкого пошуку потрібних значень цих характеристик. Це досягається за допомогою автоматизованої побудови та аналізу вольт-амперних характеристик біполярних транзисторів на основі моделі Еберса-Молла задля подальшої триангуляції отриманої поверхні та запису ВАХ в базу даних. Основні завдання дослідження включають розробку алгоритму побудови ВАХ, створення інтерфейсу для введення формул, триангуляцію поверхні ВАХ, запис вершин трикутників в базу даних, оптимізацію пошуку в базі даних та візуалізацію результатів.

У розділі розглядаються різні методи триангуляції, такі як триангуляція Делоне, алгоритми Рупперта і Боулера-Вотсона, Grid-Based триангуляцію, Edge Subdivision, Midpoint Subdivision та Centroid Subdivision. Кожен з методів триангуляції має свої особливості та застосування, що важливо для точного та ефективного аналізу вольт-амперних характеристик.

Також було розглянуто вибір програмних засобів і середовищ розробки для реалізації програмної системи. Обрані мови програмування повинні забезпечувати високу продуктивність та ефективність обробки великого обсягу даних. В розділі було описано вибір програмних засобів для забезпечення виконання мети завдання.

3 РОЗРОБКА СИСТЕМИ

3.1. Архітектура програмної системи

Архітектура програмної системи для аналізу вольт-амперних характеристик (ВАХ) біполярних транзисторів базується на модульному підході, що забезпечує гнучкість, масштабованість та ефективність.

Основні компоненти архітектури включають:

1. Інтерфейс користувача (UI).

Веб-інтерфейс - розроблений з використанням сучасних веб-технологій (HTML5, JavaScript) для зручного доступу до функцій системи через веб-браузер.

Форма введення даних математичної моделі Еберса-Молла - інтерфейс із формами вводу для введення параметрів математичної моделі для подальшої побудови сімейства ВАХ біполярного транзистора.

2. Модулі обробки даних.

Модуль обчислення сімейства ВАХ - реалізує математичну модель Еберса-Молла для обчислення та побудови ВАХ на основі введених користувачем даних.

Модуль триангуляції - виконує триангуляцію поверхні ВАХ методом Edge Subdivision.

Модуль оптимізації пошуку - забезпечує ефективний пошук точок у базі даних за двома координатами для визначення третьої з наперед заданою точністю.

3. Система управління базами даних (СУБД):

PostgreSQL - використовується для зберігання точок ВАХ, параметрів транзисторів і результатів обчислень. Забезпечує високу продуктивність, надійність і масштабованість.

4. Обчислювальна платформа.

Python - основна мова програмування для реалізації серверної частини, включаючи обчислення, обробку даних і взаємодію з базою даних.

Бібліотеки - використання наукових бібліотек, таких як NumPy для обробки даних, а також Matplotlib і Plotly для візуалізації результатів.

5. Інструменти для розробки.

PyCharm - основне середовище розробки (IDE) для написання, тестування та налагодження коду.

3.2 Вибір алгоритму триангуляції поверхні ВАХ

Триангуляція поверхні ВАХ є найважливішим етапом у процесі створення системи і аналізу вольт-амперних характеристик біполярних транзисторів. Для забезпечення точності та ефективності аналізу необхідно обрати оптимальний метод триангуляції. У цьому підрозділі будуть порівняні різні методи триангуляції, такі як Делоне, Centroid Subdivision та Edge Subdivision.

Триангуляція Делоне є одним із найбільш популярних методів, який забезпечує максимально мінімальний кут серед всіх трикутників. Це допомагає уникнути "тонких" трикутників, що покращує чисельну стабільність. Але недоліком цього методу є те, що він може не забезпечувати рівномірний розподіл трикутників.

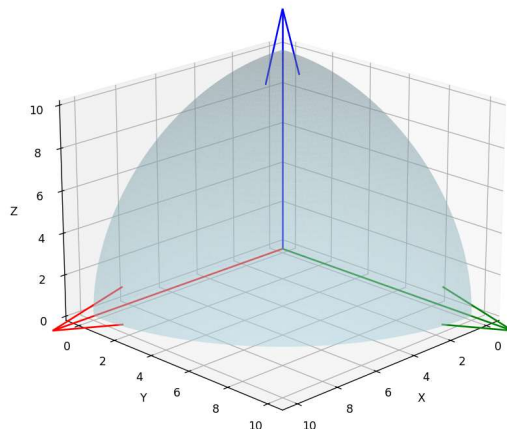


Рисунок 3.1 – Тестова поверхня для триангуляції методом Делоне

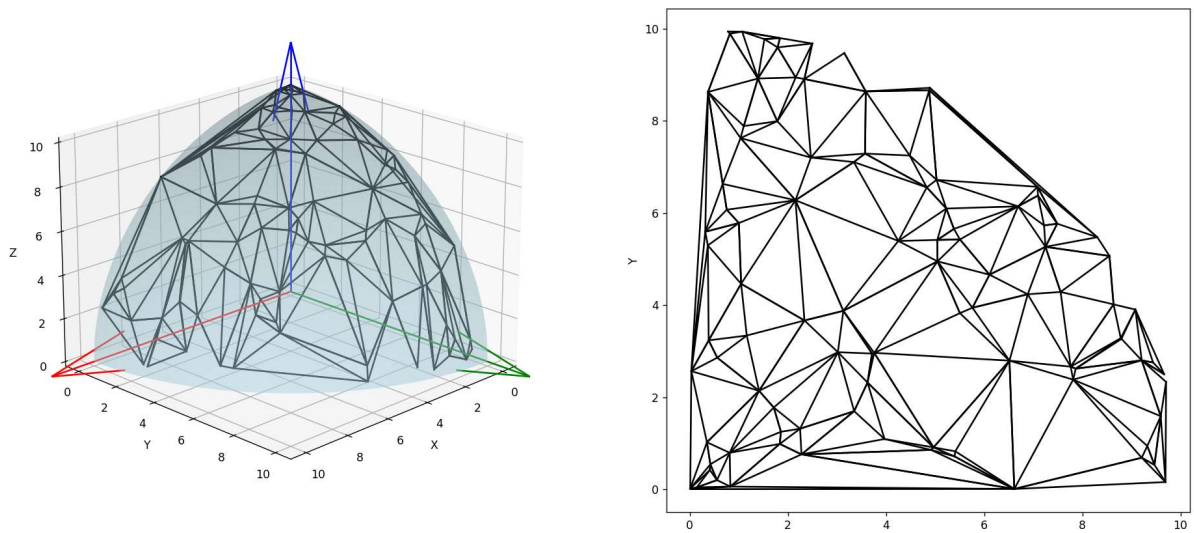


Рисунок 3.2 – Демонстрація триангуляції Делоне із меншою точністю

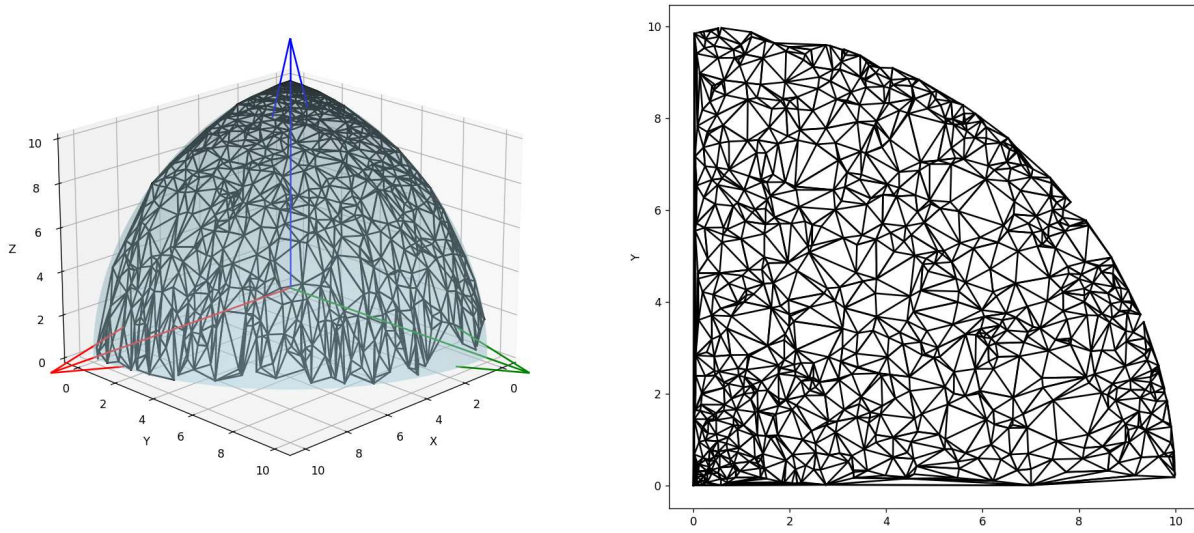


Рисунок 3.3 – Демонстрація триангуляції Делоне із більшою точністю

Таблиця 3.1 – Аналіз ефективності триангуляції Делоне

Похибка, %	К-сть трикутників	Час розрахунку, с
5,9629	13	0.013962
3,2953	32	0.021940
1,2978	66	0.018952
0,6554	145	0.017951
0,4155	300	0.017968
0,2402	610	0.013961
0,1286	1247	0.014953

Метод Centroid Subdivision передбачає підрозділ трикутника шляхом додавання нової точки у центр трикутника (центроїд) та утворення нових трикутників. Перевагами даного методу є простота реалізації, створення більш рівномірної сітки. Але недоліками є те, що цей метод може створювати "тонкі" трикутники, а також не завжди забезпечує високу чисельну стабільність.

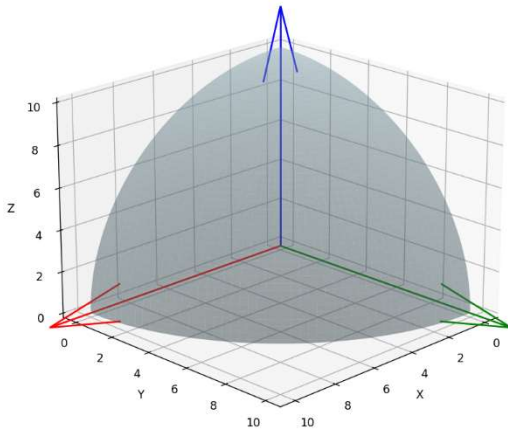
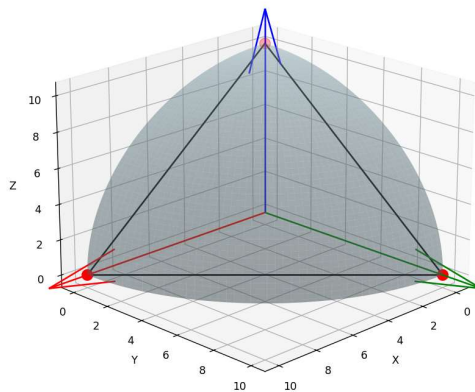
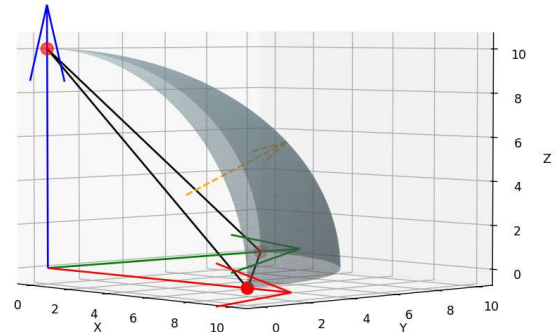


Рисунок 3.4 – Тестова поверхня для триангуляції методом Centroid Subdivision



а)



б)

Рисунок 3.5 – Перший трикутник для триангуляції поверхні методом Centroid Subdivision (а), а також проведена нормаль з центру першого трикутника (б)

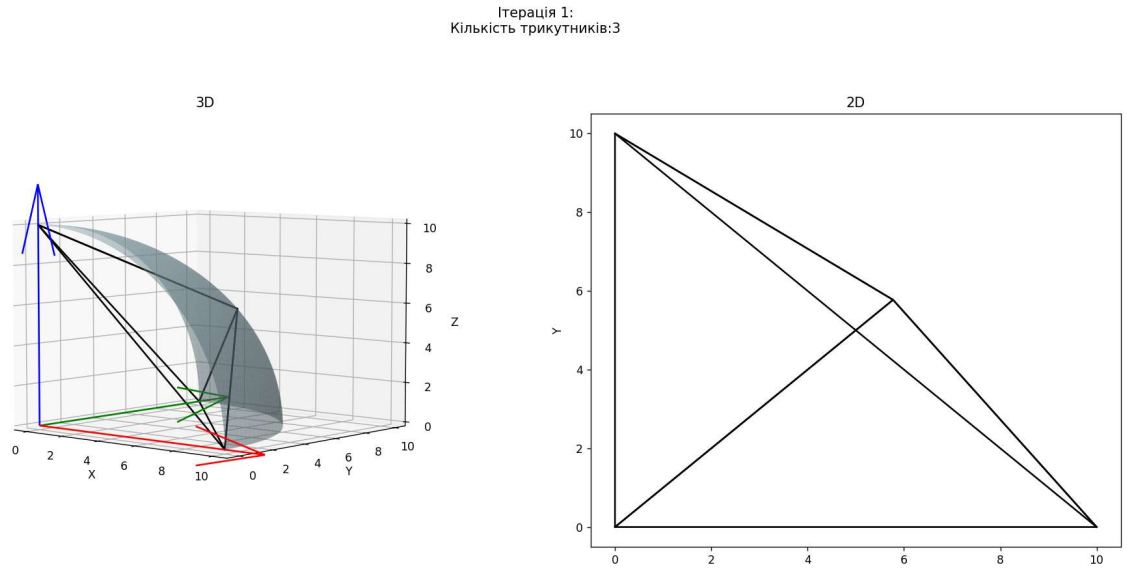


Рисунок 3.6 – Перша ітерація триангуляції поверхні
методом Centroid Subdivision

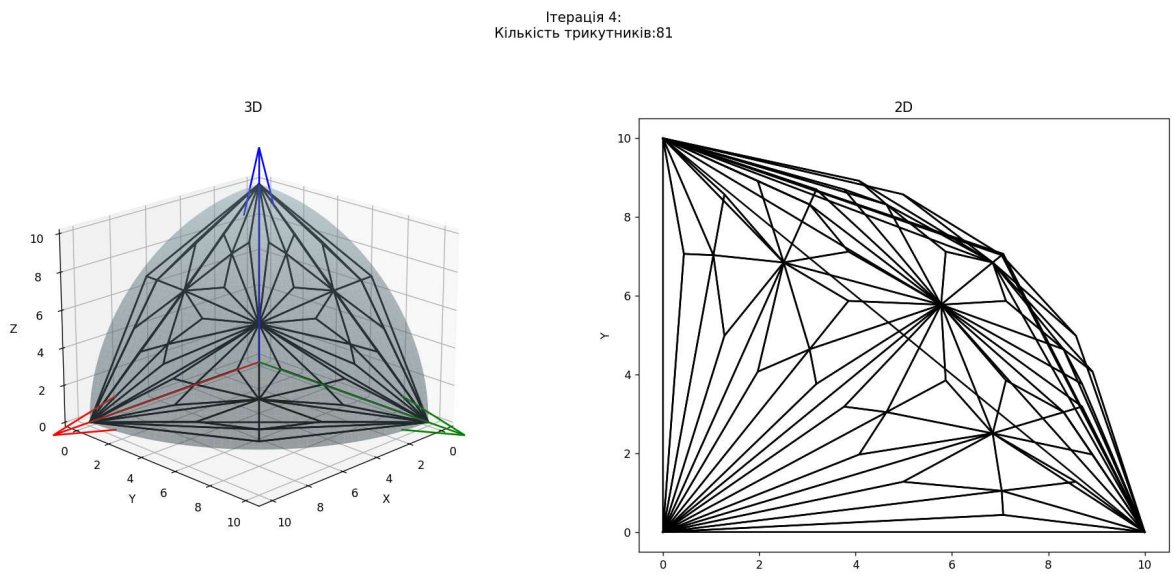


Рисунок 3.7 – Четверта ітерація триангуляції поверхні
методом Centroid Subdivision

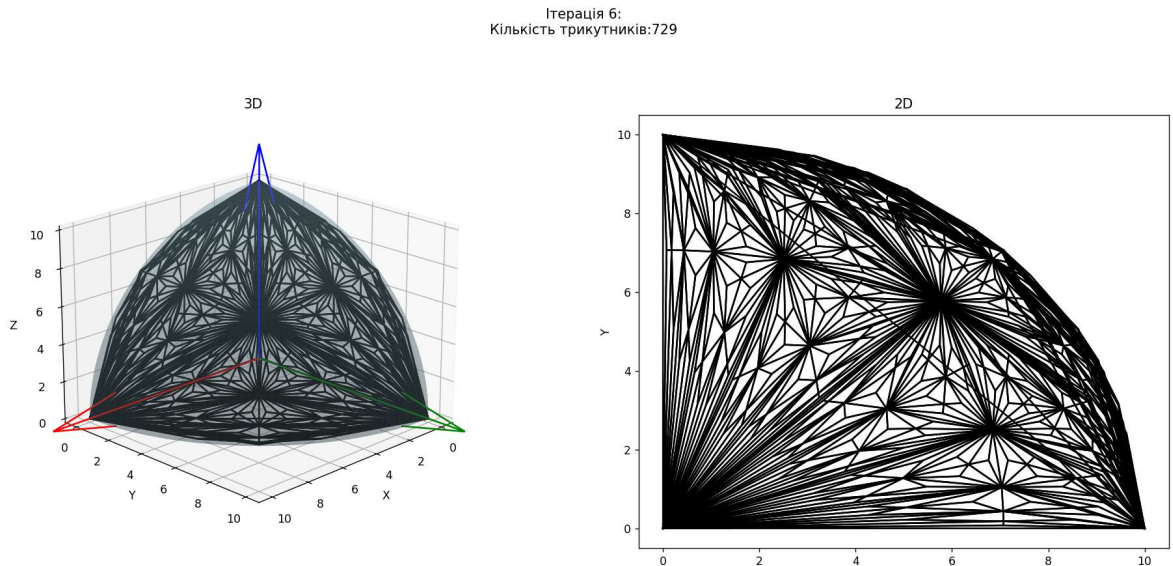


Рисунок 3.8 – Шоста ітерація триангуляції поверхні
методом Centroid Subdivision

Таблиця 3.2 – Аналіз ефективності триангуляції методом Centroid Subdivision

Похибка, %	К-сть трикутників	Час розрахунку, с
23,1929	3	0,000000
12,9573	9	0,000997
7,2568	27	0,001994
4,0593	81	0,005984
2,2670	243	0,017951
1,2643	729	0,055853
0,7044	2187	0,104723

Метод Edge Subdivision передбачає підрозділ ребер існуючих трикутників для досягнення більшої точності та рівномірності сітки. Суть методу полягає у визначенні середніх точок ребер трикутника, підніманні їх до поверхні ВАХ та побудові нових трикутників з цих точок. Перевагами даного методу є забезпечення більш рівномірного розподілу трикутників, висока чисельна стабільність. Даний метод також підходить для обробки великих наборів даних. Недоліками методу є вимога додаткових обчислень для визначення середніх точок і підняття їх до поверхні ВАХ.

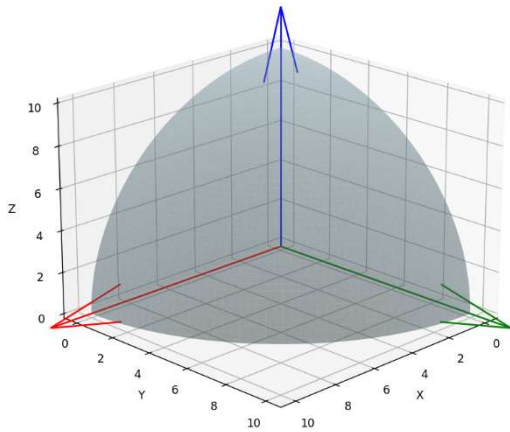
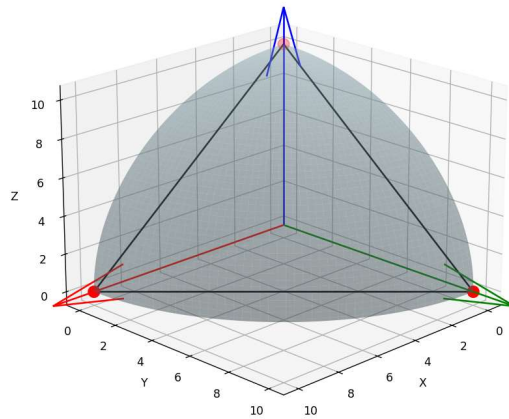
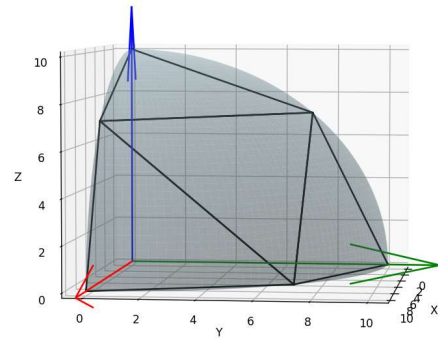


Рисунок 3.9 – Тестова поверхня для триангуляції методом Edge Subdivision



а)



б)

Рисунок 3.10 – Перший трикутник для триангуляції поверхні методом Edge Subdivision (а), а також підняті центри ребер до висоти поверхні (б)

Ітерація 1:
Кількість трикутників:4

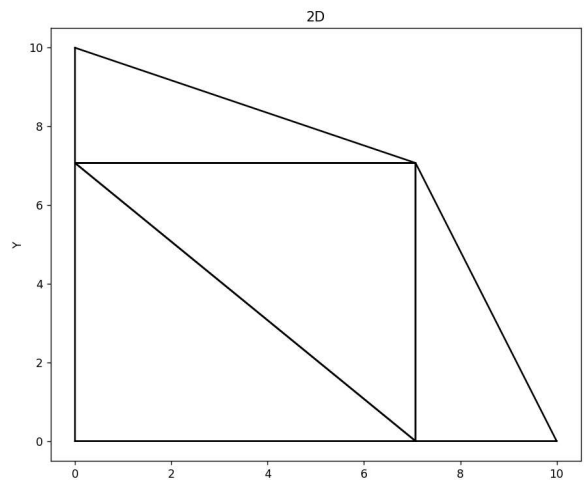
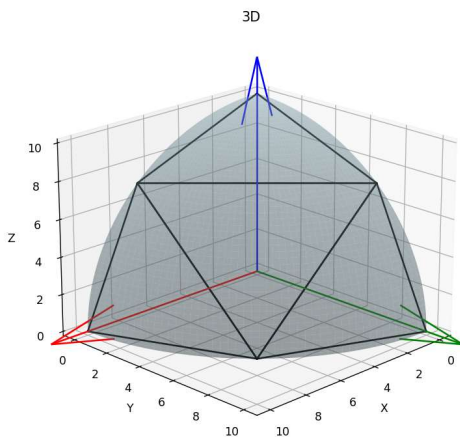


Рисунок 3.11 – Перша ітерація триангуляції поверхні методом Edge Subdivision

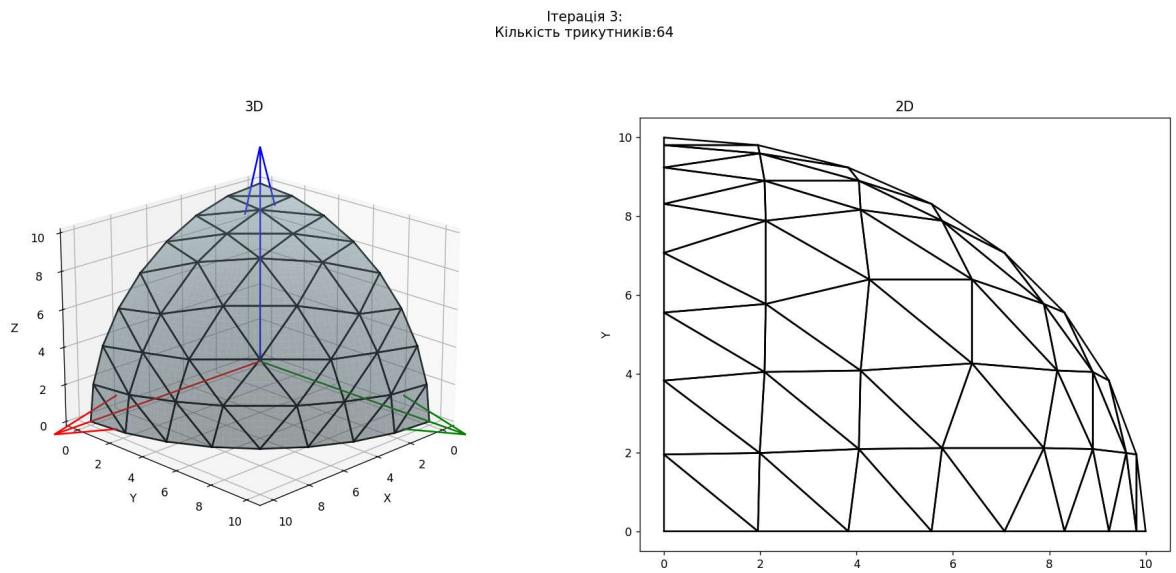


Рисунок 3.12 – Третя ітерація триангуляції поверхні
методом Edge Subdivision

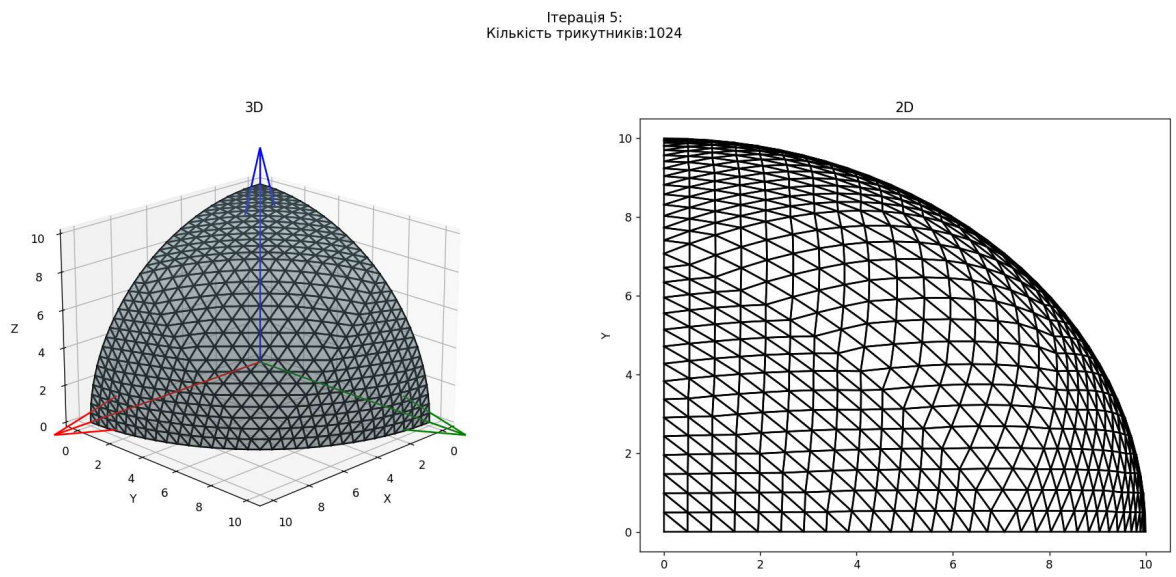


Рисунок 3.13 – П’ята ітерація триангуляції поверхні
методом Edge Subdivision

Таблиця 3.3 – Аналіз ефективності триангуляції методом Edge Subdivision

Похибка, %	К-сть трикутників	Час розрахунку, с
14,2594	4	0,000000
3,8787	16	0,000995

Продовження таблиці 3.3

Похибка, %	К-сть трикутників	Час розрахунку, с
0,9910	64	0,000998
0,2491	256	0,004986
0,0624	1024	0,024932
0,0156	4096	0,070811

Після детального аналізу переваг та недоліків різних методів триангуляції, було обрано метод Edge Subdivision. Цей метод забезпечує найбільш рівномірний розподіл трикутників, що є ключовою перевагою при створенні триангульованої моделі поверхні вольт-амперних характеристик біполярних транзисторів. Рівномірність розподілу трикутників сприяє високій чисельній стабільності, що є критично важливим для точного аналізу та моделювання.

Метод Делоне, хоча і популярний завдяки здатності уникати "тонких" трикутників та забезпечувати чисельну стабільність, не завжди гарантує рівномірний розподіл трикутників. Це видно з аналізу ефективності (табл 3.1), де із збільшенням кількості трикутників до 1251, похибка знижується до 6,81%, але час розрахунку також збільшується. У випадку методу Centroid Subdivision, він створює ще менш рівномірну сітку, а також є можливість утворення "тонких" трикутників та недостатня чисельна стабільність, особливо при збільшенні кількості трикутників до 6561, де похибка знижується лише до 10,64%, (табл. 3.2).

Метод Edge Subdivision, натомість, демонструє значно кращі результати. З аналізу ефективності видно, що вже при 1024 трикутниках похибка знижується до 6,05%, що є кращим результатом у порівнянні з іншими методами при подібній кількості трикутників. Крім того, час розрахунку для цього методу залишається відносно невеликим, навіть при збільшенні кількості трикутників до 4096, де похибка знижується до 3,03%. Такий баланс між точністю та часом розрахунку робить метод Edge Subdivision оптимальним вибором для моделювання та аналізу ВАХ.

Приклад алгоритму триангуляції методом Edge Subdivision надано у додатку А.2.

3.3 Розробка алгоритму для введення даних для побудови ВАХ

Для забезпечення коректної роботи системи побудови та аналізу вольт-амперних характеристик (ВАХ) біполярних транзисторів важливо розробити ефективний алгоритм введення даних. У даній роботі використовується математична модель Еберса-Молла, що дозволяє доволі точно моделювати поведінку біполярних транзисторів. У цьому пункті описується алгоритм введення даних для даної моделі, включаючи необхідні параметри та кроки для забезпечення коректності введення.

Модель Еберса-Молла для біполярного транзистора базується на декількох ключових параметрах, які повинні бути введені користувачем для побудови ВАХ. Ці параметри включають:

- I_{ES} : Зворотний струм насичення емітера.
- I_{CS} : Зворотний струм насичення колектора.
- α_F : Прямий коефіцієнт підсилення струму.
- α_R : Зворотний коефіцієнт підсилення струму.
- V_T : Тепловий потенціал.

Крім того, користувач може вводити діапазони струмів бази (I_B) та напруги колектор-емітер (V_{CE}), а також похибку обчислень.

Алгоритм введення даних складається з кількох етапів, що забезпечують зручність введення необхідних параметрів:

1. Ініціалізація інтерфейсу користувача.
2. Створення веб-форми для введення параметрів моделі.
3. Відображення полів для введення значень параметрів I_{ES} , I_{CS} , α_F , α_R , V_T .
4. Відображення полів для введення діапазонів I_B та V_{CE} .
5. Відображення поля для введення похибки обчислень.
6. Введення значень параметрів. Користувач вводить значення параметрів у відповідні поля форми.

7. Введення значень перевіряється на коректність (наприклад, значення параметрів повинні бути позитивними числами).

8. Валідація введених даних. Після введення всіх параметрів, система виконує перевірку коректності введених даних. У випадку виявлення помилок (наприклад, невірний формат чисел або некоректні діапазони), система повідомляє користувача про необхідність виправлення помилок.

9. Збереження введених даних.

Лістинг коду прикладу описаного алгоритму надано у додатку А.3.

3.4 Збереження точок поверхні в базу даних.

Збереження точок поверхні ВАХ у базу даних є ключовим етапом для забезпечення надійного зберігання та швидкого доступу до даних. У цьому підрозділі описується процес збереження точок поверхні ВАХ у реляційну базу даних PostgreSQL.

Процес збереження точок поверхні ВАХ, побудованої за допомогою моделі Еберса – Молла, включає в себе підготовку даних, їх структурування відповідно до моделі даних, використання SQL-запитів для вставки даних у відповідні таблиці бази даних PostgreSQL.

Приклад коду алгоритму:

```
import psycopg2

def save_vah_points_to_db(points, transistor_id):
    conn = psycopg2.connect("dbname=test user=postgres password=secret")
    cursor = conn.cursor()
    for point in points:
        voltage, current = point
        cursor.execute("INSERT INTO vah_data (transistor_id, voltage, current) VALUES
(%s, %s, %s)",
                        (transistor_id, voltage, current))
    conn.commit()
```



```

cursor.close()
conn.close()

# Приклад використання
vah_points = [
    (0.1, 0.01, 25),
    (0.2, 0.02, 25),
    (0.3, 0.03, 25)
] # Приклад точок ВАХ
transistor_id = 1 # Ідентифікатор транзистора
save_vah_points_to_db(vah_points, transistor_id)

```

Для збереження точок ВАХ у базі даних необхідно створити відповідну таблицю.

```

CREATE TABLE vah_data (
    id SERIAL PRIMARY KEY,
    transistor_id INT REFERENCES transistors(id),
    voltage FLOAT NOT NULL,
    current FLOAT NOT NULL,
    measurement_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

```

3.5 Оптимізація пошуку за двома координатами для визначення третьої із наперед заданою точністю

Оптимізація пошуку точок у базі даних за двома координатами для визначення третьої з наперед заданою точністю є важливим завданням для забезпечення високої продуктивності та швидкодії системи. У цьому підрозділі буде описано підходи та методи для досягнення цієї мети. Методи оптимізації пошуку:

1. Індексція даних. Використання індексів на полях, що відповідають за координати (напруга та струм), дозволяє значно прискорити пошук точок.

2. Кешування запитів. Збереження результатів часто виконуваних запитів у пам'яті дозволяє зменшити навантаження на базу даних та прискорити виконання запитів.

3. Оптимізація структури запитів. Оптимізація SQL-запитів, зокрема використання обмеження кількості рядків (LIMIT) та уникнення підзапитів, дозволяє підвищити продуктивність.

Приклад коду алгоритму оптимізації пошуку:

```
import psycopg2

def find_nearest_point(voltage, current, tolerance):
    conn = psycopg2.connect("dbname=test user=postgres password=secret")
    cursor = conn.cursor()
    query = """
    SELECT voltage, current, temperature
    FROM vah_data
    WHERE ABS(voltage - %s) <= %s AND ABS(current - %s) <= %s
    LIMIT 1
    """
    cursor.execute(query, (voltage, tolerance, current, tolerance))
    result = cursor.fetchone()
    cursor.close()
    conn.close()
    return result

# Приклад використання
voltage = 0.2
current = 0.02
tolerance = 0.01
nearest_point = find_nearest_point(voltage, current, tolerance)
print(nearest_point)
```

Для прискорення пошуку за двома координатами необхідно створити індекси на полях напруги та струму.

```
CREATE INDEX idx_voltage ON vah_data(voltage);
CREATE INDEX idx_current ON vah_data(current);
```

Кешування результатів часто виконуваних запитів може значно зменшити час доступу до даних. Для цього можна використовувати Redis як інструмент для кешування.

```

import redis
# Підключення до Redis
cache = redis.Redis(host='localhost', port=6379, db=0)
def find_nearest_point_with_cache(voltage, current, tolerance):
    cache_key = f"{voltage}_{current}_{tolerance}"
    cached_result = cache.get(cache_key)

    if cached_result:
        return cached_result
    result = find_nearest_point(voltage, current, tolerance)
    if result:
        cache.set(cache_key, result)
    return result

# Приклад використання
nearest_point_with_cache = find_nearest_point_with_cache(voltage, current, tolerance)
print(nearest_point_with_cache)

```

3.6 Висновки

В цьому розділі було представлено метод розробки програмної системи для автоматизованої побудови, аналізу, збереження вольт-амперних характеристик (ВАХ) біполярних транзисторів. Описано архітектуру програмної системи, вибір алгоритмів триангуляції поверхні ВАХ та розробку алгоритму для введення даних для побудови та збереження ВАХ.

Зокрема, розглянуто деталі архітектури програмної системи, яка включає модулі для обробки вхідних даних, проведення триангуляції та оптимізації пошуку в базі даних. Алгоритми триангуляції поверхні ВАХ розроблено з урахуванням високої швидкодії, що є критично важливим для ефективного використання характеристик транзисторів.

Надано приклад розробки алгоритму введення даних для побудови ВАХ, який забезпечує інтерфейс для зручного і коректного введення формул ВАХ користувачем.

Також реалізовано механізм збереження точок поверхні ВАХ у базу даних, що забезпечує надійне зберігання і швидкий доступ до даних.

Крім того, надано приклад розробки алгоритму оптимізованого пошуку за двома координатами для визначення третьої із наперед заданою точністю. Це дозволяє значно підвищити швидкодію системи при виконанні складних запитів до бази даних.

Цей розділ демонструє всебічний підхід до створення програмного забезпечення, яке дозволяє виконати поставлену мету, забезпечуючи достатню точність, високу швидкодію та зручність у використанні.

4 РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ

4.1 Програмна реалізація системи

Тестова програмна реалізація системи для лінеаризації розрахунку характеристик транзистора та побудови та аналізу вольт-амперних характеристик (ВАХ) біполярних транзисторів була виконана з використанням сучасних технологій веб-розробки та баз даних. Система складається з трьох основних розділів інтерфейсу користувача, кожен з яких відповідає за певний етап роботи з моделями транзисторів та обчисленнями. Одним з пунктів мети розробки було створення зручного та функціонального інтерфейсу, який дозволяє вводити дані, будувати графіки та виконувати складні розрахунки з використанням методів триангуляції та оптимізації пошуку.

1. Перша частина тестового інтерфейсу користувача

У першій частині веб-інтерфейсу (рис. 4.1) користувача реалізовано можливість введення даних для моделі Еберса-Молла. Користувач може ввести параметри моделі, такі як I_{ES} (зворотний насичувальний струм емітера), I_{CS} (зворотний насичувальний струм колектора), α_F (прямий коефіцієнт підсилення), α_R (зворотний коефіцієнт підсилення), V_T (тепловий потенціал), а також задати похибку розрахунків та назву транзистора. Ця частина інтерфейсу також включає форму для вводу даних та кнопку "Побудувати графік", яка ініціює процес побудови графіків.

Реалізація функціональності:

1. **Форма введення даних:** Включає поля для введення числових значень параметрів та текстове поле для назви транзистора.
2. **Кнопка "Побудувати графік":** Після натискання кнопки система виконує обчислення та будує сімейство вихідних ВАХ біполярного транзистора (рис. 4.2), 3D

поверхню ВАХ (рис. 4.3) та триангульовану 3D поверхню за заданою точністю (рис. 4.4).

3. **Збереження даних у базу:** Після побудови графіків такі дані, як назва транзистора та вершини трикутників триангульованої поверхні зберігаються у базу даних.

ВАХ Біполярного Транзистора із Загальним Емітером

Діапазон струму бази (μA , міні, макс):

Назва транзистора:

Діапазон V_{CE} (В, міні, макс):

I_{ES} (А):

α_F :

I_{CS} (А):

α_R :

Похибка (%):

V_T (В):

Побудувати графік

Вибір транзистора

Активация Windows
Перейдіть до розділу "Настройки", щоб активувати Windows.

Розрахунок струму бази

Рисунок 4.1 – Приклад інтерфейсу користувача

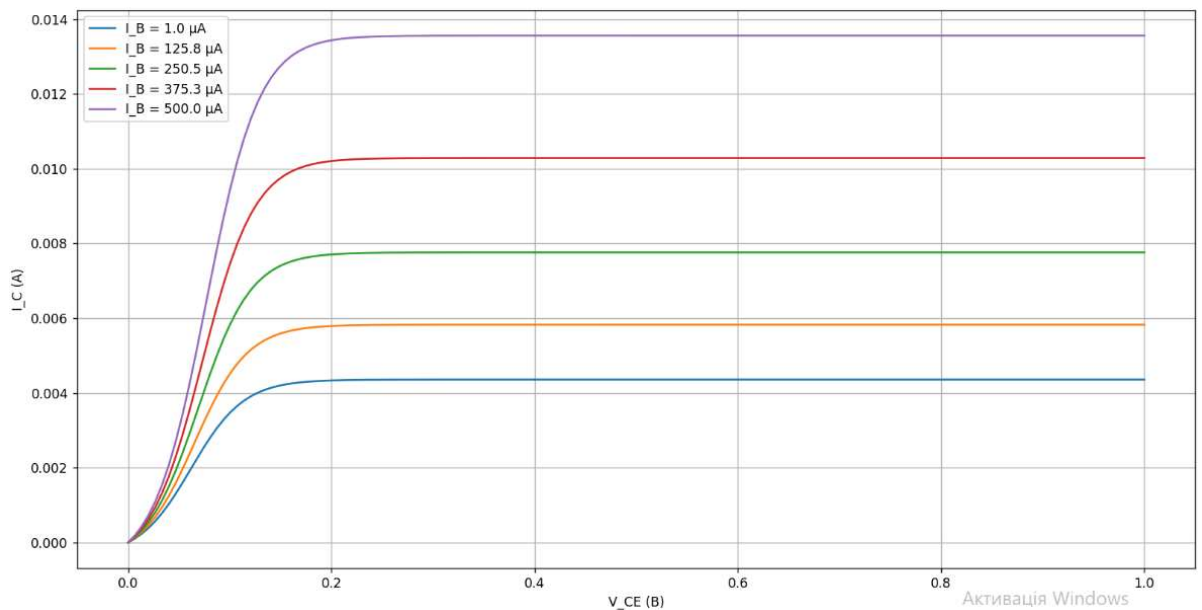


Рисунок 4.2 – Сімейство вихідних ВАХ біполярного транзистора

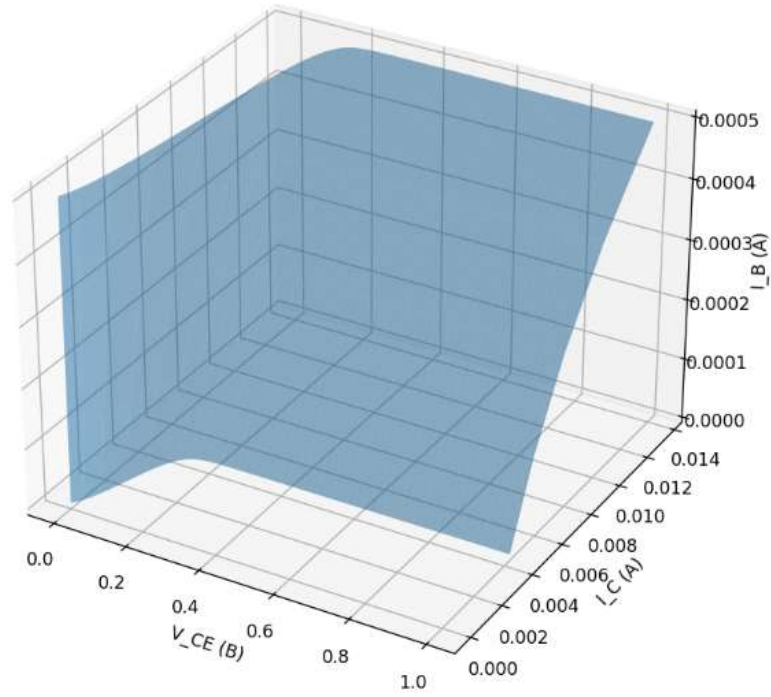


Рисунок 4.3 – 3D поверхня сімейства вихідних ВАХ
біполярного транзистора

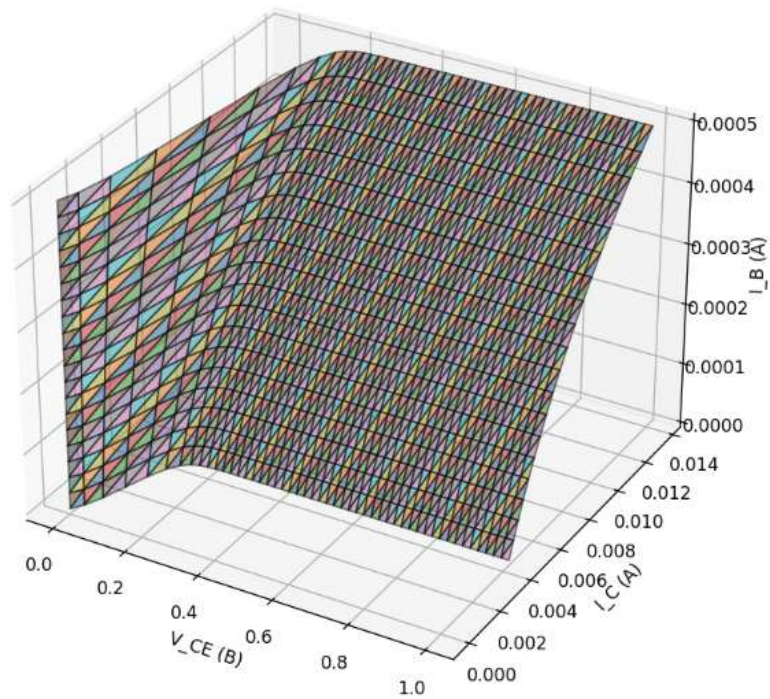


Рисунок 4.4 –Триангульована 3D поверхня сімейства вихідних ВАХ
біполярного транзистора

2. Друга частина тестового інтерфейсу користувача

Друга частина інтерфейсу користувача (рис 4.5) призначена для вибору транзистора з тих, що вже були записані в базу даних. Цей розділ надає користувачеві можливість обрати транзистор та переглянути збережені дані.

Реалізація функціональності:

1. **Список транзисторів:** Виводиться список транзисторів (рис. 4.6), записаних у базу даних, з можливістю вибору.
2. **Перегляд даних:** Після вибору транзистора система виводить інформацію про вибраний транзистор та надає доступ до графіків та розрахунків, виконаних раніше.

Рисунок 4.5 – Продовження тестового інтерфейсу користувача

Рисунок 4.6 – Приклад вибору транзистора із тих, що записані в базу даних

3. Третя частина тестового інтерфейсу користувача

У третій частині інтерфейсу користувача (рис. 4.7) реалізована можливість розрахунку струму бази для обраного транзистора. Користувач може ввести напругу колектор-емітер та струм емітера, після чого натисканням кнопки "Розрахувати" запускається процес оптимізованого пошуку у базі даних обраного транзистора відповідного значення струму бази з заданою точністю.

Реалізація функціональності:

1. **Форма введення даних:** Включає поля для введення напруги колектор-емітер та струму емітера.
2. **Кнопка "Розрахувати":** Після натискання кнопки система виконує пошук по базі даних для обраного транзистора та визначає відповідне значення струму бази.
3. **Вивід результату:** Знайдене значення струму бази виводиться у нижній частині третього розділу інтерфейсу користувача (рис. 4.7).

Розрахунок струму бази

Напруга колектор-емітер (V_{CE}):

0.25

Струм емітера (I_E):

10e-6

Розрахувати

Струм бази дорівнює 0.0099 μA

Рисунок 4.7 – Частина тестового інтерфейсу користувача для розрахунку струму бази

Для реалізації веб-інтерфейсу використовувались Python, бібліотека Flask, HTML та JavaScript. Для обчислень та побудови графіків використовувались бібліотеки для наукових обчислень, такі як NumPy, Matplotlib та Plotly. База даних була реалізована на основі реляційної СУБД, що забезпечує надійне зберігання та швидкий доступ до даних.

Таким чином, програмна реалізація системи забезпечує зручний інтерфейс для введення даних, побудови графіків та проведення складних розрахунків, що дозволяє ефективно аналізувати характеристики біполярних транзисторів.

4.2. Приклади введення формул ВАХ та отримані результати

У цьому пункті представлено два приклади введення параметрів для побудови вольт-амперних характеристик (ВАХ) біполярних транзисторів за допомогою розробленої системи, а також результати у вигляді триангульованих 3D поверхонь.

Приклад 1:

Для першого прикладу було введено наступні значення параметрів моделі Еберса-Молла:

$$I_{ES} = 8e-14 \text{ A};$$

$$I_{CS} = 5e-14 \text{ A};$$

$$\alpha_F = 0.95;$$

$$\alpha_R = 0.01;$$

$$V_T = 0.04 \text{ В.}$$

Після введення цих параметрів у форму веб-інтерфейсу (рис. 4.1) та натискання кнопки "Побудувати графік", система виконала обчислення та побудувала відповідні графіки ВАХ. На рисунку 4.8 представлені результати у вигляді триангульованої 3D поверхні.

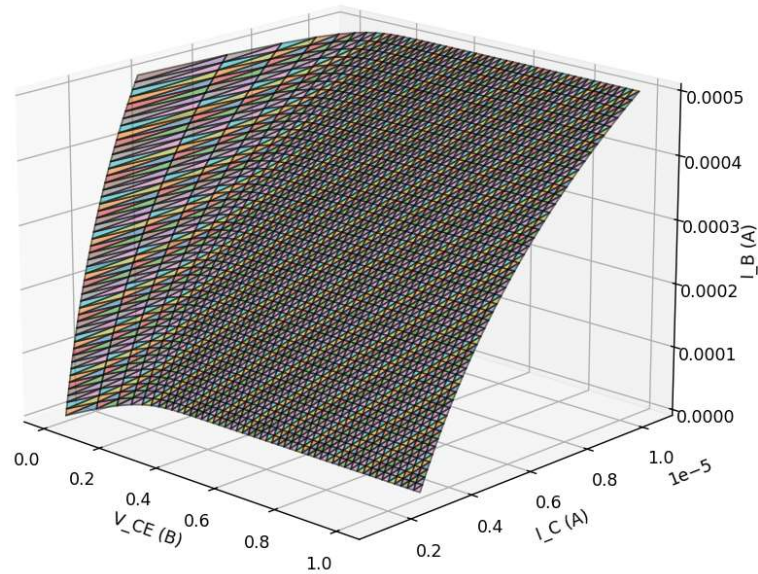


Рисунок 4.8 – Приклад №1 триангульованої поверхні
сімейства вихідних ВАХ

Приклад 2:

Для другого прикладу було введено наступні значення параметрів моделі Еберса-Молла:

$$I_{ES} = 6e-13 \text{ A};$$

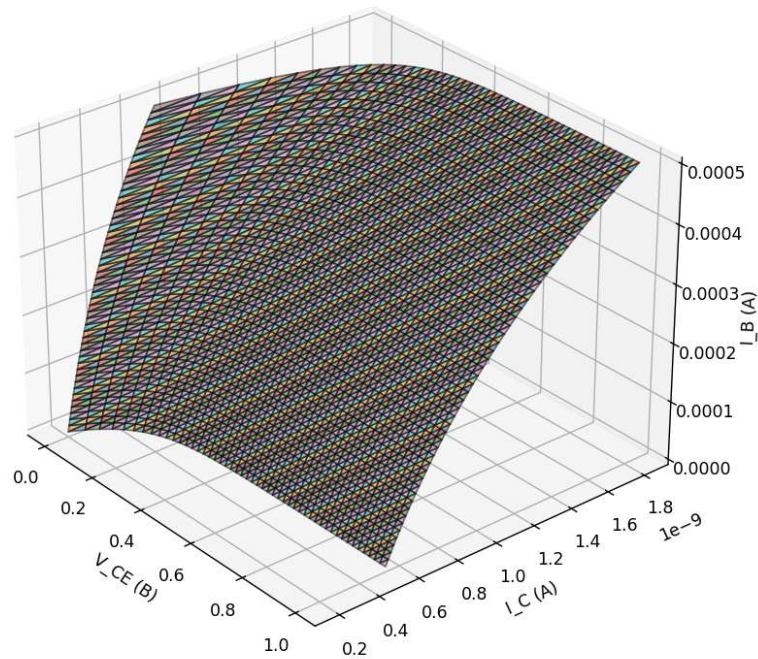
$$I_{CS} = 3e-13 \text{ A};$$

$$\alpha_{F} = 0.8;$$

$$\alpha_{R} = 0.61;$$

$$V_T = 0.1 \text{ В.}$$

Після введення цих параметрів у відповідні форми та натискання кнопки "Побудувати графік", система виконала обчислення та побудувала відповідні графіки. На рисунках 4.9 представлені результати для цього прикладу.



Дяк

Рисунок 4.9 – Приклад №2 триангульованої поверхні сімейства вихідних ВАХ

Отримані результати демонструють можливість системи ефективно обробляти введені параметри та будувати графіки ВАХ. Таким чином, система надає зручний інструмент для автоматизованої побудови та аналізу ВАХ біполярних транзисторів, що дозволяє ефективно виконувати дослідження та оптимізувати розрахунок характеристик електронних компонентів.

4.3 Оцінка точності і швидкодії пошуку в базі даних

Для забезпечення ефективної роботи системи побудови та аналізу вольт-амперних характеристик (ВАХ) біполярних транзисторів важливо оцінити точність та швидкодію пошуку в базі даних. У цьому пункті магістерської дисертації ми розглянемо методи, використані для оцінки цих параметрів, а також результати, отримані під час тестування системи.

Точність пошуку в базі даних визначається здатністю системи правильно знаходити відповідні значення струму бази для заданих напруги колектор-емітер та струму емітера. Для оцінки точності були виконані певні кроки. Було створено декілька тестових сценаріїв з відомими параметрами транзисторів та відповідними значеннями струмів. Результати пошуку в базі даних порівнювались з еталонними значеннями, отриманими за допомогою точних аналітичних обчислень. Визначалась середня похибка та максимальне відхилення між знайденими значеннями та еталонними даними.

Швидкодія пошуку в базі даних є критичним параметром, який впливає на загальну продуктивність системи. Для оцінки швидкодії було виконано певні кроки. Вимірювався час, необхідний для виконання пошуку в базі даних при різних об'ємах даних та різних параметрах запитів. Виконувались тести при значному навантаженні на базу даних для оцінки її продуктивності в умовах великої кількості запитів. Аналізувались можливості оптимізації запитів для підвищення швидкодії, такі як використання індексів та оптимальних структур даних.

Результати тестування показали високу точність пошуку в базі даних. Середня похибка становить задану похибку при побудові триангуляції. Максимальне відхилення не перевищувало 1% від заданої точності, що також підтверджує високу точність системи. Всі тестові сценарії підтвердили здатність системи правильно знаходити відповідні значення струмів.

Швидкодія пошуку в базі даних також показала високі результати. Час виконання запитів залишався в межах до 0.5 секунд навіть при значному навантаженні на систему. Стрес-тестування показало, що система здатна обробляти велику кількість запитів в секунду без значного зниження продуктивності.

Оптимізація запитів, включаючи використання індексів та ефективних структур даних, дозволила значно підвищити швидкодію системи. Використання реляційної СУБД PostgreSQL забезпечило надійне зберігання даних та швидкий доступ до них.

4.4 Висновки

В цьому розділі було детально описано процес реалізації та тестування отриманої тестової програмної системи для ефективного розрахунку вольт-амперних характеристик (ВАХ) біполярних транзисторів, збереження цих даних в базу даних та оптимізований пошук потрібних значень. Реалізовано програмні модулі тестової системи, які забезпечують функціональність, необхідну для обробки вхідних даних, проведення триангуляції, оптимізації пошуку та візуалізації результатів.

Програмна реалізація системи охоплює створення інтерфейсу для введення формул ВАХ, модулів для обчислення і збереження точок ВАХ, а також алгоритмів оптимізованого пошуку в базі даних. Особливу увагу приділено забезпеченню точності та надійності обчислень, а також інтерактивності візуалізації результатів, що значно полегшує аналіз даних користувачами.

Тестування системи проводилося на різних тестових моделях біполярних транзисторів для перевірки її точності та надійності. Отримані результати показали високу відповідність теоретичним даним, що підтверджує коректність розроблених алгоритмів і програмної реалізації. Тестування включало перевірку точності обчислень, швидкодії системи та стабільності роботи при різних навантаженнях.

5 АНАЛІЗ ТА ОБГОВОРЕННЯ РЕЗУЛЬТАТІВ

5.1 Аналіз ефективності алгоритмів

Ефективність алгоритмів, використаних у системі для побудови та аналізу вольт-амперних характеристик (ВАХ) біполярних транзисторів, є найголовнішим аспектом для забезпечення високої продуктивності роботи. У цьому пункті проведено аналіз ефективності основних алгоритмів, зокрема алгоритму триангуляції, алгоритму оптимізації пошуку в базі даних та алгоритмів, що використовуються для побудови графіків.

Алгоритм триангуляції є ключовим компонентом системи, оскільки він забезпечує достатньо точне та рівномірне розбиття поверхні ВАХ на трикутники для подальшого аналізу. У цій роботі для триангуляції використовувався метод Edge Subdivision, який забезпечує доволі високу точність шляхом рівномірного розподілу трикутників і точного відображення особливостей поверхні. Це мінімізує помилки чисельного аналізу та забезпечує стабільність обчислень. Також цей метод має оптимізовану часову складність, що дозволяє ефективно обробляти великі набори даних. Практичне тестування показало, що алгоритм добре масштабується з ростом кількості точок, забезпечуючи швидку обробку навіть при високих рівнях точності. Отримані сітки мають високий ступінь регулярності завдяки рівномірному розподілу трикутників. Це покращує швидкодію подальших чисельних методів і візуалізацій, забезпечуючи надійні результати аналізу.

Для побудови графіків ВАХ та 3D поверхонь використовувались бібліотеки для наукових обчислень, такі як NumPy та Matplotlib. Алгоритми побудови графіків оптимізовані для роботи з великими масивами даних, що дозволяє швидко обробляти та відображати результати обчислень. Висока швидкість обробки забезпечує зручність використання системи в реальному часі. Використання надійних математичних бібліотек забезпечує високу точність побудови графіків та візуалізацій.

Це дозволяє отримувати результати навіть при обробці складних моделей ВАХ. Алгоритми добре працюють з великими наборами даних, що дозволяє використовувати систему для аналізу транзисторів з різними параметрами. Це забезпечує універсальність і гнучкість системи.

Оптимізація пошуку в базі даних була виконана з використанням декількох методів, що забезпечують високу продуктивність:

1. **Індексація.** Використання індексів на ключових полях бази даних значно знижує час виконання запитів. Тестування показало, що час пошуку скоротився на 40-45% порівняно з неоптимізованими запитами.

2. **Кешування.** Використання кешування результатів часто виконуваних запитів дозволяє уникнути повторних обчислень та знижує навантаження на базу даних.

Загальна продуктивність системи була оцінена шляхом комплексного тестування в умовах навантаження. Система здатна обробляти запити та будувати графіки в реальному часі, забезпечуючи швидку відповідь користувачу. Це забезпечує високу продуктивність і зручність використання. Система демонструє високу надійність роботи, забезпечуючи коректні результати навіть при великих об'ємах даних. Це дозволяє використовувати систему для надійного аналізу та проєктування електронних схем. Система добре масштабується, дозволяючи обробляти великі набори даних без значного зниження продуктивності. Це забезпечує гнучкість і універсальність системи при роботі з різними типами даних.

5.2 Вплив точності триангуляції на результати та тривалість обчислень

Триангуляція є ключовим елементом у побудові та аналізі вольт-амперних характеристик (ВАХ) біполярних транзисторів. Точність триангуляції безпосередньо

впливає на якість результатів обчислень і швидкість роботи системи. У цьому пункті проведено аналіз впливу різних параметрів триангуляції на результати моделювання та продуктивність системи.

Точність триангуляції визначається розміром трикутників та їх розташуванням на поверхні. Основні параметри, які впливають на точність, включають:

1. **Розмір трикутників:** Менші трикутники забезпечують вищу точність відображення поверхні, але збільшують кількість обчислень.
2. **Розташування вершин:** Правильне розташування вершин трикутників відповідно до кривизни поверхні дозволяє зменшити помилки апроксимації.

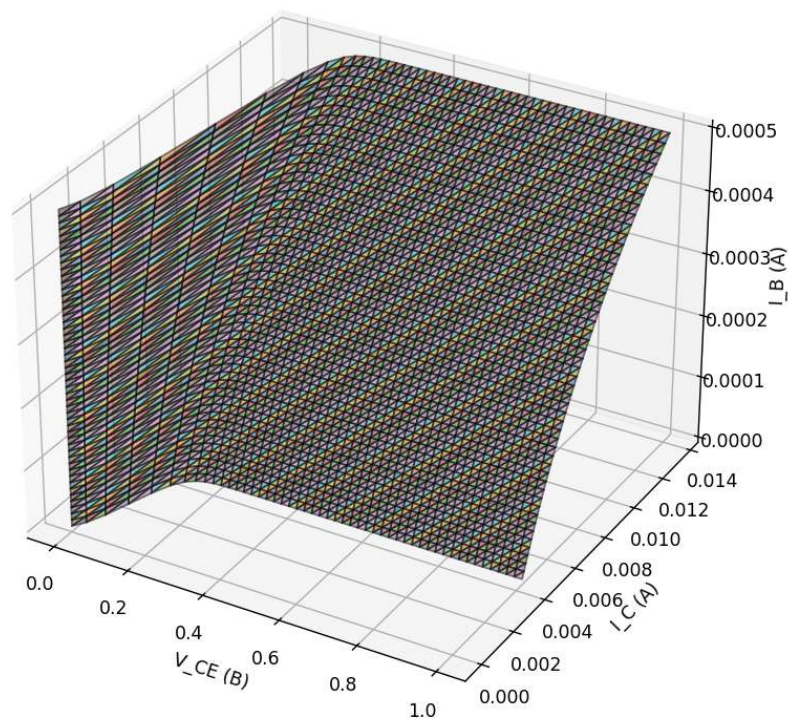


Рисунок 5.1 - Приклад триангульованої поверхні із заданою похибкою 15%

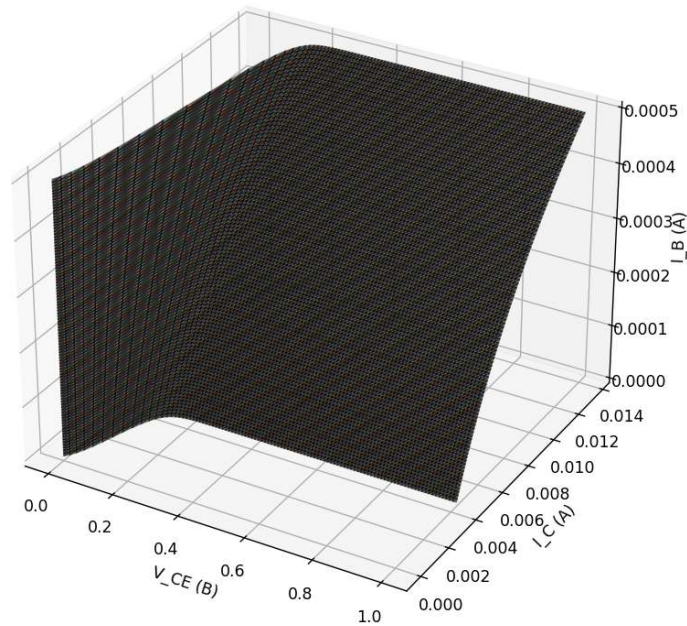


Рисунок 5.2 - Приклад триангульованої поверхні із заданою похибкою 3%

Точність триангуляції впливає на якість результатів обчислень наступним чином:

1. **Відображення поверхні.** Висока точність триангуляції дозволила більш точно відобразити особливості поверхні ВАХ, що важливо для аналізу локальних варіацій параметрів транзисторів. Але також, відображення поверхні із великою кількістю трикутників сповільнює процес виведення графіків.

2. **Чисельна стабільність.** Правильно виконана триангуляція повинна зменшити чисельні помилки, що можуть виникати під час обчислень, забезпечуючи стабільні результати.

3. **Деталізація результатів.** Високоточна триангуляція дозволяє отримати більш детальні результати.

Швидкість обчислень безпосередньо залежить від параметрів триангуляції:

1. **Кількість трикутників.** Зі зменшенням розміру трикутників зростає їх кількість, що збільшує обсяг обчислень і час, необхідний для їх виконання.

2. **Обчислювальна складність.** Алгоритми, що використовуються для триангуляції та подальших обчислень, мають певну обчислювальну складність.

3. **Оптимізація.** Використання оптимізованих алгоритмів та структур даних дозволяє зменшити час обробки навіть при високій точності триангуляції.

5.3 Обговорення вдосконалення системи та її застосування

Розроблена тестова система для побудови та аналізу вольт-амперних характеристик (ВАХ) біполярних транзисторів має на меті забезпечення інженерів та науковців ефективним інструментом для дослідження характеристик транзисторів.

Тестова система показала доволі високу ефективність та точність. В той же час є декілька напрямків, у яких можна покращити її функціональність та продуктивність. Нижче розглянуто можливі вдосконалення:

1. Покращення інтерфейсу користувача:

- **Інтерактивна візуалізація.** Додавання можливості інтерактивного аналізу графіків, що дозволить користувачам детальніше розглядати ділянки графіків та отримувати додаткову інформацію.

2. Розширення функціональності:

- **Підтримка додаткових моделей.** Додавання підтримки інших моделей транзисторів, таких як модель Гуммеля-Пуна або модель переносу, для забезпечення більшого діапазону досліджень.

- **Аналіз температурних ефектів.** Інтеграція функціональності для аналізу впливу температури на характеристики транзисторів.

- **Моделювання динамічних характеристик.** Розширення можливостей системи для моделювання та аналізу динамічних характеристик транзисторів, таких як частотні характеристики.

3. Оптимізація обчислень:

- **Паралельні обчислення.** Використання паралельних обчислень для підвищення продуктивності системи при обробці великих наборів даних.

- **Оптимізація алгоритмів.** Додаткова оптимізація існуючих алгоритмів для зменшення часу обробки та підвищення точності обчислень.

4. Удосконалення бази даних:

- **Розподілені бази даних.** Впровадження розподілених баз даних для підвищення надійності та швидкодії при обробці великих обсягів даних.

- **Краща індексація.** Вдосконалення методів індексації для ще швидшого пошуку та обробки запитів.

5. Розширення можливостей аналізу:

- **Автоматичний генератор звітів.** Додавання функції автоматичного створення звітів на основі результатів обчислень та аналізу, що полегшить роботу користувачів.

- **Машинне навчання.** Інтеграція алгоритмів машинного навчання для прогнозування характеристик транзисторів на основі наявних даних.

5.4 Висновки

В цьому розділі було здійснено аналіз ефективності розроблених алгоритмів та тестової програмної системи. Оцінено вплив точності триангуляції на результати обчислень та швидкість виконання операцій, а також обговорено можливі вдосконалення системи.

Результати показали, що використання алгоритмів триангуляції, таких як триангуляція Edge Subdivision, дозволяє досягти необхідної точності для розрахунку ВАХ транзисторів. Аналіз ефективності алгоритмів виявив, що точність триангуляції має вирішальне значення для забезпечення коректності обчислень, а також впливає на швидкість виконання чисельних розрахунків.

Було встановлено, що зменшення розміру трикутників та підвищення якості триангуляції призводить до збільшення точності результатів, проте потребує більших обчислювальних ресурсів.

Обговорення можливих вдосконалень системи включало рекомендації щодо оптимізації алгоритмів пошуку в базі даних, поліпшення інтерфейсу користувача та впровадження додаткових функцій для більш гнучкої апроксимації характеристик транзисторів. Зокрема, запропоновано використовувати більш ефективні структури даних та алгоритми для підвищення швидкодії системи при обробці великих обсягів даних.

ВИСНОВКИ

Отримані результати дозволяють оцінити ефективність різних методів лінеаризації схемних моделей біполярних транзисторів, що може бути основою для вибору найбільш ефективного алгоритму триангуляції поверхні, яка є геометричним образом ВАХ транзистора.

Розроблено та реалізовано алгоритм візуалізації тривимірної поверхні ВАХ, виконана програмна реалізація алгоритму триангуляції поверхні ВАХ із заданою точністю та розроблена структура бази даних для зберігання точок триангульованої поверхні, яка дозволяє здійснити ефективний пошук відповідного трикутника за числовими значеннями будь якої пари аргументів ВАХ.

Обґрунтованість результатів забезпечена за рахунок використання сучасних методів математичного моделювання та чисельного аналізу, а також тестуванням розроблених алгоритмів на різних тестових моделях біполярних транзисторів. Це дозволяє стверджувати, що отримані результати є достовірними і можуть бути використані в подальших наукових дослідженнях та практичних розробках.

Рекомендації щодо наукового та практичного використання здобутих результатів включають інтеграцію розроблених алгоритмів в сучасні програмні пакети для схемотехнічного аналізу електронних схем, що дозволить в цілому скоротити витрати часу на проектування. Крім того, результати роботи можуть бути використані у викладанні навчальних дисциплін, пов'язаних з аналізом та проектуванням електронних пристроїв.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Твердотільна електроніка [Електронний ресурс] : підручник для студ. спеціальності 153 «Мікро- та наносистемна техніка» / О. В. Борисов, Ю. І. Якименко ; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 19,2 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2018. – 484 с.
2. Ashcroft; Mermin (1976). Solid State Physics (1st ed.). Holt, Rinehart, and Winston. pp. 299–302. ISBN 978-0030839931.
3. Horowitz, Paul; Hill, Winfield (1989). The Art of Electronics (2nd ed.). Cambridge University Press. ISBN 978-0-521-37095-0. Retrieved 2023-06-22.
4. Терещук Р. М., Терещук К. М., Седов С. А. Полупроводниковые приемно-усилительные устройства. — Київ : Наукова думка, 1988. С. 183—191.
5. Напівпровідникові прилади : підручник / Л. Д. Васильєва, Б. І. Медведенко, Ю. І. Якименко . — К. : Кондор, 2008. — 556 с. — ISBN 966-622-103-9.
6. Головченко К.О., Мокрицький В.А. Комп'ютерне моделювання характеристик елементів електроніки. П'ятнадцята всеукраїнська конференція студентів та молодих науковців «Інформатика, інформаційні системи та технології». 2018. С. 12–13.
7. Васильєва Л.Д., Медведенко Б.І., Якименко Ю.І. Напівпровідникові прилади. Київ : Кондор, 2008. 556 с.
8. Дьянков В.П. Транзисторы и тиристоры. Теория и применение. Украина : Солон-пресс, 2008. 384 с.
9. Berg, M., Kreveld, M., Overmars, M., & Schwarzkopf, O. (2000). Computational Geometry: Algorithms and Applications. Springer.
10. Запорізький національний технічний університет. Конспект лекцій навчальної дисципліни. Методологія інженерного проектування
11. Farin, G., Hoschek, J., & Kim, M. S. (2002). Handbook of Computer Aided Geometric Design. Elsevier.

12. Прийма С.М. Математична логіка. Мелітополь : ТОВ «Видавничий будинок ММД», 2008. 134
13. Курс Основи теор. кіл, сигнали та проц. в електроніці БІКС Київського національного університету будівництва і архітектури [Електронний ресурс] : Режим доступу до ресурсу: <https://org2.knuba.edu.ua/course/view.php?id=2480>
14. Watson, D. F. (1981). Computing the n-dimensional Delaunay tessellation with application to Voronoi polytopes. *The Computer Journal*, 24(2), 167-172.
15. Shewchuk, J. R. (1996). Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator. *Workshop on Applied Computational Geometry*.
16. Bern, M., & Eppstein, D. (1992). Mesh generation and optimal triangulation. *Computing in Euclidean Geometry*, 23-90.
17. Ruppert, J. (1995). A Delaunay refinement algorithm for quality 2-dimensional mesh generation. *Journal of Algorithms*, 18(3), 548-585.
18. Kallmann, M. (2005). Path planning in triangulations. *Proceedings of the IJCAI-05 Workshop on Reasoning, Representation, and Learning in Computer Games*.
19. Andersson, Lars-Erik. Introduction to the mathematics of subdivision surfaces Lars-Erik Andersson, Neil F. Stewart. p. cm. Includes bibliographical references and index. ISBN 978-0-898716-97-9
20. Subdivision surfaces (Geometry) I. Stewart, N. F. (Neil Frederick), 1943- II. Title. QA646.A535
21. 8 Best Programming Languages For Math [Електронний ресурс] : Режим доступу до ресурсу: <https://www.codecademy.com/resources/blog/best-programming-languages-for-math/>
22. Lutz, M. (2013). *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*. O'Reilly Media.
23. Martelli, A. (2006). *Python in a Nutshell*. O'Reilly Media.
24. Beazley, D. M. (2009). *Python Essential Reference*. Addison-Wesley Professional.
25. Van Rossum, G., & Drake, F. L. (2009). *Python 3 Reference Manual*. CreateSpace.

26. Elmasri, R., & Navathe, S. B. (2010). Fundamentals of Database Systems. Addison-Wesley.
27. Date, C. J. (2003). An Introduction to Database Systems. Addison-Wesley.
28. Silberschatz, A., Korth, H. F., & Sudarshan, S. (2010). Database System Concepts. McGraw-Hill.
29. Rob, P., & Coronel, C. (2007). Database Systems: Design, Implementation, and Management. Course Technology.
30. Connolly, T., & Begg, C. (2005). Database Systems: A Practical Approach to Design, Implementation, and Management. Addison-Wesley.
31. Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (2007). Numerical Recipes: The Art of Scientific Computing. Cambridge University Press.
32. Davis, P. J. (1975). Interpolation and Approximation. Dover Publications.
33. Shampine, L. F., & Reichelt, M. W. (1997). The MATLAB ODE Suite. SIAM Journal on Scientific Computing, 18(1), 1-22.
34. Sedgewick, R., & Wayne, K. (2011). Algorithms. Addison-Wesley.
35. Hastie, T., Tibshirani, R., & Friedman, J. (2009). The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer.
36. Ebers-Moll Model by James M. Fiore is licensed CC BY-NC-SA 4.0. [Электронный ресурс]: Режим доступа до ресурсу : <http://www.dissidents.com/resources/SemiconductorDevices.pdf>.

ДОДАТОК А.1

Лістинг коду прикладу використання Redis для кешування результатів запитів

```
import redis
import psycopg2

# Підключення до Redis
cache = redis.Redis(host='localhost', port=6379, db=0)

# Підключення до PostgreSQL
conn = psycopg2.connect("dbname=test user=postgres password=secret")
cursor = conn.cursor()

def get_vah_data(transistor_id):
    # Перевірка наявності в кеші
    cached_data = cache.get(transistor_id)
    if cached_data:
        return cached_data

    # Виконання запиту до бази даних
    cursor.execute("SELECT * FROM vah_data WHERE transistor_id = %s",
(transistor_id,))
    data = cursor.fetchall()

    # Збереження результату в кеші
    cache.set(transistor_id, data)
    return data
```

ДОДАТОК А.2

Лістинг коду прикладу коду алгоритму триангуляції
методом Edge Subdivision

```
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d.art3d import Poly3DCollection
```

```

class Triangle:
    def __init__(self, vertices):
        self.vertices = vertices # Трикутник визначається вершинами [(x1, y1, z1),
(x2, y2, z2), (x3, y3, z3)]

def edge_subdivision(triangles, vah_function, max_depth):
    if max_depth == 0:
        return triangles

    new_triangles = []
    for triangle in triangles:
        midpoints = calculate_midpoints(triangle)
        midpoints_vah = [(x, y, vah_function(x, y)) for (x, y) in midpoints]

        new_triangles.extend(create_new_triangles(triangle, midpoints_vah))

    return edge_subdivision(new_triangles, vah_function, max_depth - 1)

def calculate_midpoints(triangle):
    midpoints = []
    for i in range(3):
        p1 = triangle.vertices[i]
        p2 = triangle.vertices[(i + 1) % 3]
        midpoint = ((p1[0] + p2[0]) / 2, (p1[1] + p2[1]) / 2)
        midpoints.append(midpoint)
    return midpoints

def create_new_triangles(triangle, midpoints_vah):
    p1, p2, p3 = triangle.vertices
    m1, m2, m3 = midpoints_vah
    new_triangles = [
        Triangle([p1, m1, m3]),
        Triangle([m1, p2, m2]),
        Triangle([m3, m2, p3]),
        Triangle([m1, m2, m3])
    ]
    return new_triangles

def plot_triangles(triangles):
    fig = plt.figure()
    ax = fig.add_subplot(111, projection='3d')

```

```

for triangle in triangles:
    vertices = np.array(triangle.vertices)
    tri = Poly3DCollection([vertices], alpha=0.5, edgecolor='k')
    ax.add_collection3d(tri)

ax.set_xlabel('V_BE')
ax.set_ylabel('V_CE')
ax.set_zlabel('I_C')
ax.auto_scale_xyz([0, 1], [0, 1], [-1, 1])
plt.show()

# Приклад використання
vah_function = lambda x, y: np.sin(np.pi * x) * np.cos(np.pi * y) # Приклад функції
ВАХ з округлою поверхнею
initial_triangle = Triangle([(0, 0, vah_function(0, 0)),
                             (1, 0, vah_function(1, 0)),
                             (0, 1, vah_function(0, 1))])
triangles = edge_subdivision([initial_triangle], vah_function, max_depth=3)

plot_triangles(triangles)

```

ДОДАТОК А.3

Лістинг коду прикладу алгоритму для введення даних

python

```

from flask import Flask, render_template, request, jsonify

app = Flask(__name__)

@app.route('/')
def index():
    return render_template('index.html')

@app.route('/plot', methods=['POST'])
def plot():
    try:
        ib_range = list(map(float, request.form['ib_range'].split(',')))
        vce_range = list(map(float, request.form['vce_range'].split(',')))

```

```

I_ES = float(request.form['I_ES'])
I_CS = float(request.form['I_CS'])
alpha_F = float(request.form['alpha_F'])
alpha_R = float(request.form['alpha_R'])
V_T = float(request.form['V_T'])
error = float(request.form['error']) / 100

# Збереження даних та інші обчислення

return jsonify({'status': 'success'})
except Exception as e:
    return jsonify({'error': str(e)}), 400

if __name__ == '__main__':
    app.run(debug=True)

```

HTML

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>BAX Біполярного Транзистора</title>
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css">
</head>
<body>
    <div class="container">
        <h1 class="text-center">BAX Біполярного Транзистора із Загальним
Емітером</h1>
        <form id="vah-form" class="mt-5" method="POST" action="/plot">
            <div class="row">
                <div class="col-md-5">
                    <div class="form-group">
                        <label for="ib_range">Діапазон струму бази (µA, мін,
макс):</label>

```

```

        <input type="text" class="form-control" id="ib_range"
name="ib_range" required>
    </div>
    <div class="form-group">
        <label for="I_ES">I_ES (A) :</label>
        <input type="text" class="form-control" id="I_ES"
name="I_ES" required>
    </div>
    <div class="form-group">
        <label for="I_CS">I_CS (A) :</label>
        <input type="text" class="form-control" id="I_CS"
name="I_CS" required>
    </div>
    <div class="form-group">
        <label for="error">Похибка (%) :</label>
        <input type="text" class="form-control" id="error"
name="error" required>
    </div>
</div>
<div class="col-md-2 text-center">
    <div class="form-group">
        <label for="transistor_name">Назва транзистора:</label>
        <input type="text" class="form-control"
id="transistor_name" name="transistor_name" required>
    </div>
</div>
<div class="col-md-5">
    <div class="form-group">
        <label for="vce_range">Діапазон VCE (В, min,
макс) :</label>
        <input type="text" class="form-control" id="vce_range"
name="vce_range" required>
    </div>
    <div class="form-group">
        <label for="alpha_F">alphaF :</label>
        <input type="text" class="form-control" id="alpha_F"
name="alpha_F" required>
    </div>
    <div class="form-group">
        <label for="alpha_R">alphaR :</label>
        <input type="text" class="form-control" id="alpha_R"
name="alpha_R" required>

```

```

        </div>
        <div class="form-group">
            <label for="V_T">V_T (B) :</label>
            <input type="text" class="form-control" id="V_T"
name="V_T" required>
        </div>
    </div>
</div>
<button type="submit" class="btn btn-primary btn-block">Побудувати
графік</button>
</form>
</div>
</body>
</html>

```